If Interpretability Is the Answer, What Is the Question? – A Causal Perspective

Gunnar König

Dissertation an der
Fakultät für Mathematik, Informatik und Statistik der
Ludwig–Maximilians–Universität München

München, den 26. Juli 2023

First Co-Supervisor: Prof. Moritz Grosse-Wentrup
Research Group Neuroinformatics
Department of Computer Science
Universität Wien

Second Co-Supervisor: Prof. Bernd Bischl
Chair for Statistical Learning and Data Science
Institute for Statistics
Ludwig–Maximilians–Universität München

First Reviewer: Prof. Bernd Bischl
Second Reviewer: Prof. Thomas Augustin
External Reviewer: Prof. Zachary Lipton

Date of Defense: 20th of October 2023

Summary

Due to the ability to model even complex dependencies, machine learning (ML) can be used to tackle a broad range of (high-stakes) prediction problems. The complexity of the resulting models comes at the cost of transparency, meaning that it is difficult to understand the model by inspecting its parameters. This opacity is considered problematic since it hampers the transfer of knowledge from the model, undermines the agency of individuals affected by algorithmic decisions, and makes it more challenging to expose non-robust or unethical behaviour.

To tackle the opacity of ML models, the field of interpretable machine learning (IML) has emerged. The field is motivated by the idea that if we could understand the model's behaviour – either by making the model itself interpretable or by inspecting post-hoc explanations – we could also expose unethical and non-robust behaviour, learn about the data generating process, and restore the agency of affected individuals. IML is not only a highly active area of research, but the developed techniques are also widely applied in both industry and the sciences.

Despite the popularity of IML, the field faces fundamental criticism, questioning whether IML actually helps in tackling the aforementioned problems of ML and even whether it should be a field of research in the first place: First and foremost, IML is criticised for lacking a clear goal and, thus, a clear definition of what it means for a model to be interpretable. On a similar note, the meaning of existing methods is often unclear, and thus they may be misunderstood or even misused to hide unethical behaviour. Moreover, estimating conditional-sampling-based techniques poses a significant computational challenge.

With the contributions included in this thesis, we tackle these three challenges for IML.

We join a range of work by arguing that the field struggles to define and evaluate ”interpretability” because incoherent interpretation goals are conflated. However, the different goals can be disentangled such that coherent

requirements can inform the derivation of the respective target estimands. We demonstrate this with the examples of two interpretation contexts: recourse and scientific inference.
To tackle the misinterpretation of IML methods, we suggest deriving formal interpretation rules that link explanations to aspects of the model and data. In our work, we specifically focus on interpreting feature importance. Furthermore, we collect interpretation pitfalls and communicate them to a broader audience.
To efficiently estimate conditional-sampling-based interpretation techniques, we propose two methods that leverage the dependence structure in the data to simplify the estimation problems for Conditional Feature Importance (CFI) and SAGE.

A causal perspective proved to be vital in tackling the challenges: First, since IML problems such as algorithmic recourse are inherently causal; Second, since causality helps to disentangle the different aspects of model and data and, therefore, to distinguish the insights that different methods provide; And third, algorithms developed for causal structure learning can be leveraged for the efficient estimation of conditional-sampling based IML methods.

Zusammenfassung

Aufgrund der Fähigkeit, selbst komplexe Abhängigkeiten zu modellieren, kann maschinelles Lernen (ML) zur Lösung eines breiten Spektrums von anspruchsvollen Vorhersageproblemen eingesetzt werden. Die Komplexität der resultierenden Modelle geht auf Kosten der Interpretierbarkeit, d. h. es ist schwierig, das Modell durch die Untersuchung seiner Parameter zu verstehen. Diese Undurchsichtigkeit wird als problematisch angesehen, da sie den Wissenstransfer aus dem Modell behindert, sie die Handlungsfähigkeit von Personen, die von algorithmischen Entscheidungen betroffen sind, untergräbt und sie es schwieriger macht, nicht robustes oder unethisches Verhalten aufzudecken.

Um die Undurchsichtigkeit von ML-Modellen anzugehen, hat sich das Feld des interpretierbaren maschinellen Lernens (IML) entwickelt. Dieses Feld ist von der Idee motiviert, dass wir, wenn wir das Verhalten des Modells verstehen könnten - entweder indem wir das Modell selbst interpretierbar machen oder anhand von post-hoc Erklärungen - auch unethisches und nicht robustes Verhalten aufdecken, über den datengenerierenden Prozess lernen und die Handlungsfähigkeit betroffener Personen wiederherstellen könnten. IML ist nicht nur ein sehr aktiver Forschungsbereich, sondern die entwickelten Techniken werden auch weitgehend in der Industrie und den Wissenschaften angewendet.

Trotz der Popularität von IML ist das Feld mit fundamentaler Kritik konfrontiert, die in Frage stellt, ob IML tatsächlich dabei hilft, die oben genannten Probleme von ML anzugehen, und ob es überhaupt ein Forschungsgebiet sein sollte: In erster Linie wird an IML kritisiert, dass es an einem klaren Ziel und damit an einer klaren Definition dessen fehlt, was es für ein Modell bedeutet, interpretierbar zu sein. Weiterhin ist die Bedeutung bestehender Methoden oft unklar, so dass sie missverstanden oder sogar missbraucht werden können, um unethisches Verhalten zu verbergen. Letztlich stellt die Schätzung von auf bedingten Stichproben basierenden Verfahren eine erhebliche rechnerische Herausforderung dar.

In dieser Arbeit befassen wir uns mit diesen drei grundlegenden Herausforderungen von IML.
Wir schließen uns der Argumentation an, dass es schwierig ist, ”Interpretierbarkeit” zu definieren und zu bewerten, weil inkohärente Interpretationsziele miteinander vermengt werden. Die verschiedenen Ziele lassen sich jedoch entflechten, sodass kohärente Anforderungen die Ableitung der jeweiligen Zielgrößen informieren. Wir demonstrieren dies am Beispiel von zwei Interpretationskontexten: algorithmischer Regress und wissenschaftliche Inferenz.
Um der Fehlinterpretation von IML-Methoden zu begegnen, schlagen wir vor, formale Interpretationsregeln abzuleiten, die Erklärungen mit Aspekten des Modells und der Daten verknüpfen. In unserer Arbeit konzentrieren wir uns speziell auf die Interpretation von sogenannten Feature Importance Methoden. Darüber hinaus tragen wir wichtige Interpretationsfallen zusammen und kommunizieren sie an ein breiteres Publikum.
Zur effizienten Schätzung auf bedingten Stichproben basierender Interpretationstechniken schlagen wir zwei Methoden vor, die die Abhängigkeitsstruktur in den Daten nutzen, um die Schätzprobleme für Conditional Feature Importance (CFI) und SAGE zu vereinfachen.

Eine kausale Perspektive erwies sich als entscheidend für die Bewältigung der Herausforderungen: Erstens, weil IML-Probleme wie der algorithmische Regress inhärent kausal sind; zweitens, weil Kausalität hilft, die verschiedenen Aspekte von Modell und Daten zu entflechten und somit die Erkenntnisse, die verschiedene Methoden liefern, zu unterscheiden; und drittens können wir Algorithmen, die für das Lernen kausaler Struktur entwickelt wurden, für die effiziente Schätzung von auf bindingten Verteilungen basierenden IML-Methoden verwenden.

Contents

Chapter 1

General Introduction

1.1 Overview

Due to the ability to model even complex dependencies, machine learning (ML) can be used to tackle a broad range of (high-stakes) prediction problems [Tarca et al., 2007, Kourou et al., 2015, Zeng et al., 2017, Wuest et al., 2016, Liakos et al., 2018, Raghavan et al., 2020]. The complexity of the resulting models comes at the cost of transparency, meaning that it is difficult to understand how the model works by inspecting its parameters.

This so-called opacity is considered problematic since our trust in ML models is based on estimating the model's risk, but there are requirements that loss functions and test sets do not capture [Doshi-Velez and Kim, 2017]: For example, models may rely on non-robust associations [Lapuschkin et al., 2019] or may pick up unfair or unethical behaviour [Bender et al., 2021]. Furthermore, individuals affected by algorithmic decisions should be able to contest and change them [Wachter et al., 2017a, Ustun et al., 2019, Freiesleben, 2021], and practitioners are interested in using the model to gain insight into the data [Doshi-Velez and Kim, 2017, Freiesleben et al., 2022].

To tackle the opacity of ML models, the field of interpretable machine learning (IML) has emerged [Breiman, 2001, Friedman, 2001, Ribeiro et al., 2016, Lundberg and Lee, 2017, Wachter et al., 2017b, Ustun et al., 2019, Covert et al., 2020, Karimi et al., 2020a, Molnar, 2020]. The field is motivated by the hope that if we understood the model's behaviour – either because the model itself was interpretable or because we had access to post-hoc explanations – we could also expose unethical and non-robust behaviour, learn about the data-generating process and restore the agency of affected individuals.

Although IML is increasingly applied in research and practice [Fellous et al., 2019, Deeks, 2019, Gade et al., 2019, Gordon et al., 2019, Danilevsky et al., 2020, Jiménez-Luna et al., 2020, Tosun et al., 2020, Das et al., 2021, Tantithamthavorn and Jiarpakdee, 2021, Sharma et al., 2022, Yang, 2022, Khosravi et al., 2022, Gevaert, 2022, Machlev et al., 2022, Fiok et al., 2022], the field falls short of expectations. IML is criticised for suffering from fundamental problems:

1. Often, the goal of IML is seen in making models ”interpretable”, but there is no clear definition of what that means [Lipton, 2018, Páez, 2019, Freiesleben and König, 2023];
2. IML methods themselves are subject to interpretation, but we lack clear interpretation rules [Krishna et al., 2022, Freiesleben and König, 2023]; and
3. many theoretically appealing methods require knowledge about the data-generating process that is usually not readily available [Hooker and Mentch, 2019, Frye et al., 2020].

The contributions in this thesis are focused on tackling these three issues.

Contributions

Towards Clarification of Interpretation Goals and Target Estimands. IML is criticized for lacking a definition of what it means for a model to be opaque or interpretable [Lipton, 2018, Páez, 2019, Freiesleben and König, 2023] — and therefore to lack the means to evaluate the proposed methods. Finding a definition for “interpretability” is impossible because the term conflates several incompatible goals [Lipton, 2018, Páez, 2019, Freiesleben and König, 2023]. To make progress, the different goals must be disentangled such that coherent requirements can motivate the design and choice of IML methods.
Throughout the thesis, we investigate two interpretation contexts: *recourse* [König et al., 2023, 2021] and *scientific inference* [Freiesleben et al., 2022]. We demonstrate that by disentangling each context from other interpretation scenarios, coherent requirements can be derived.

Towards Preventing Misinterpretation of IML. The interpretation of explanations is often unclear, and thus methods may be misunderstood or misused to hide unethical behaviour [Rudin, 2019, Krishna et al., 2022, Bordt et al., 2022, Freiesleben and König, 2023]. To tackle the misinterpretation of IML methods, we suggest deriving formal interpretation rules that link explanations to aspects of the model and data; In our work, we specifically focus on the interpretation of feature importance. Furthermore, we propose estimators to quantify the uncertainties involved in their estimation [Molnar et al., 2021]. To raise awareness in a broader audience, we collect interpretation pitfalls and illustrate them on examples [Molnar et al., 2022].

Towards efficient estimation of conditional-sampling-based methods. To be useful, IML methods must often incorporate knowledge about the data-generating process; however, this knowledge is usually not readily available and difficult and expensive to obtain.

More specifically, a range of work argues that IML methods should not evaluate the model in unseen and unrealistic regions [Hooker and Mentch, 2019, Frye et al., 2020, Chen et al., 2020, Freiesleben et al., 2022]; therefore feature perturbations must be constructed such that dependencies with the remaining features are preserved. More specifically the perturbations must be sampled from the conditional distribution of the feature given its covariates. Learning conditional samplers is a difficult and expensive task [Zhou et al., 2022].

To tackle the issue, we propose two methods that enable a more efficient estimation of conditional-sampling-based methods [Molnar et al., 2023, Luther et al., 2023].

Importance of a Causal Perspective

A causal perspective proved particularly important in tackling the three aforementioned challenges.

Firstly, many questions in IML are inherently causal and thus, causality is required to formalise and estimate the target estimands. For example, with *recourse* explanations, we aim to guide individuals rejected by an algorithmic system towards *actions* that allow them to revert the unfavourable decision; Thus, recourse recommendations are concerned with causal effects.

Secondly, many IML methods are based on perturbations of the model inputs – a form of intervention; Thus, we need causality to capture the meaning of the methods' outputs and to assess whether a link between the explanation

goal and explanation technique can be established.[1]
On a more pragmatic stance, causal structure learning methods allowed us to greedily identify conditional independencies in the data, which proved helpful in making the estimation of a conditional-sampling-based interpretation technique (SAGE values) more efficient.

Structure of the Dissertation

The dissertation is structured in four parts. The first and current part is the general introduction (§1). We started this chapter with a high-level overview of the thesis contents. As follows, we introduce our notation and background on supervised learning (§1.2.1), causality (§1.2.2), and interpretable machine learning (§1.2.3). Then, we postulate the three challenges for IML that in our view are most pressing, and that we tackled in our contributions (§1.3). As a first contribution, using the introduced notation and causal formalism, we introduce a taxonomy of nine different perspectives on model and data (§1.4), which will help us to create an explanatory link between explanation and explanandum.
After the general introduction (§1) we present the included articles (§2), summarise how our contributions tackle the postulated challenge and discuss limitations and open problems (§3), and conclude the thesis by returning to the eponymous hook: If Interpretability is the Answer, What is the Question (§4)?

1.2 Notation and Background

In this section, we introduce notation and background relevant to this thesis.

- In §1.2.1, we introduce notation for the supervised learning paradigm.
- In §1.2.2, we recapitulate causal concepts relevant to our work, including the do-operator, causal graphs, d-separation, observational identifiability, the ladder of causation and structural causal models.
- In §1.2.3, we introduce important terminology in the field of IML as well as the definitions of the IML methods most relevant to our work, i.e., Permutation Feature Importance (PFI), Conditional Feature Importance (CFI), Partial Dependence Plots (PDPs), Individual Conditional Expectation (ICE) curves, M-Plots, SHAP and SAGE values, as well as Counterfactual Explanations (CEs) and Causal Recourse (CR).

[1] IML methods are typically based on assessing the effect of perturbations of the model inputs. We will formalise such model-level interventions in §1.4.

notation	meaning
$X, Y, \hat{Y}$	features, target, prediction
$\hat{f}, h$	prediction model, raw prediction function
L, R, R_{emp}	loss, risk, empirical risk
$X \perp Y\|Z$, $X \not\perp Y\|Z$	conditional dependence, conditional independence
$do(X = x)$	setting variable X to value x with an intervention
$X \perp_d Y\|Z$, $X \not\perp_d Y\|Z$	d-separation, no d -separation
X^{pre}	pre-intervention state of X (factual)
X^{post}	post-intervention state (counterfactual)
$-j$	all features except j, i.e., $D\backslash\{j\}$
$\tilde{X}_S$, $\tilde{X}_S^C$	perturbations $\sim P(X_S)$, $P(X_S\|X_C)$
$\mathcal{G}$	causal graph (more graph notation in Table 1.2)

Table 1.1: Overview of our notation.

Readers familiar with the introduced concepts and methods may skip the background sections and may refer to Table 1.1 for a summary of our notation, and to §4 for a list of abbreviations.

1.2.1 Prediction Model and Statistical Notation

In our work, we focus on interpreting supervised machine learning (ML) models. We denote the prediction target as Y, the covariates (variables, features) as X, the estimated model as $\hat{f}$, and the corresponding prediction as $\hat{Y} := \hat{f}(X)$. For discrete targets, the raw prediction function is denoted as h. The loss function is denoted as L, the risk as R and the empirical risk as R_{emp}. We capitalise random variables and use the respective lowercase letter for observations. For example, $X = x$ means that the random variable X takes value x. We write probability distributions as $P(X)$ and event probabilities as $P(X = x)$ or $p(x)$. To indicate that X and Y are conditionally independent given Z, we write $X \perp Y|Z$ and $X \not\perp Y|Z$ to indicate their dependence. We also refer to dependence as association or co-occurrence and to linear dependence as correlation.

Supervised ML models are designed to approximate aspects of the conditional distribution of Y given the covariates X, i.e., $P(Y|X)$. Which aspect is modelled depends on the choice of the loss function. If the model is well specified and optimal w.r.t. the mean squared error, it reflects the conditional expectation $E[Y|X]$. Similarly, for categorical targets, if the model is optimal w.r.t. cross-entropy loss, it reflects $P(Y|X)$ [Hastie et al., 2009].

1.2.2 A Brief Introduction to Causal Inference

Supervised ML models excel at modelling associations; They can be used to predict within the distribution on which the model was trained. For example, we can use supervised learning models to diagnose diseases based on observations of potential causes and symptoms.
However, in many scenarios, we are not only interested in association but also in causality; and association does not imply causation.

- For example, in medicine, we do not only want to diagnose diseases but also want to treat them. These are qualitatively different questions: While disease diagnosis is only concerned with association (co-occurrence), treatment effect estimation is an inherently causal task, that cannot be answered with association alone: Although symptoms are associated with the disease, only interventions on causes have the potential to heal (while interventions on symptoms will leave the disease unaffected).
- As teased in §1.1, the distinction between association and causation matters for IML and thus for this thesis: First of all, interpretation goals such as *recourse* are concerned with causal effects. Secondly, many IML methods perform some form of intervention, such that a causal formalism is required to express their meaning.

The field of *causal inference* is concerned with formalising and answering causal questions. As follows, I introduce fundamental concepts and notation that are necessary to follow this dissertation. The contents of this section only scratch the surface of the rich body of work in causality. For a more in-depth introduction, we refer to a range of textbooks on the matter [Spirtes et al., 2000, Pearl, 2009, Peters et al., 2017, Miguel et al., 2023].

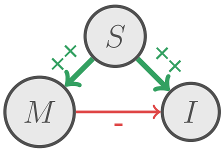


Figure 1.1: Although mosquitoes M have a negative causal impact on tourism income I, both are positively correlated. The reason is that the confounder season S has a strong positive impact on both.

Simpson's Paradox

It is well known that association does not imply causation. Simpson's paradox serves as a vivid illustration of this difference. It shows that even when two variables exhibit a positive correlation, their underlying causal relationship may be negative or non-existent. This paradox arises due to confounding variables that influence both the dependent and independent variables, inducing misleading associations. It highlights the necessity of careful causal analysis and consideration of confounders to draw accurate conclusions about causal effects.

Let us explain the paradox at an example (visualised in Figure 1.1). Suppose the Swedish government wants to understand how the prevalence of mosquitoes M impacts income from tourism I. Suppose furthermore that season S affects both mosquito prevalence and tourism income: Both tourists and mosquitoes generally prefer summertime. Because of this *confounding*, tourism income and mosquito prevalence are positively associated. If the Swedish government were to equate association with causation, they might decide to grow the mosquito population to attract more tourists. However, since tourists dislike mosquitoes, the real causal effect is the opposite.

The do-operator

In the preceding paragraphs, we illustrated that association and causation are distinct concepts. While associative statements concern the distribution of the variables in the environment in which they were observed (the *observational* distribution), causal statements involve interventions that *change* the causal dependencies in the data. Thus, they concern distributions that are, in general, different from the observational distribution. We refer to these distributions as *interventional* distributions.

The standard statistical notation allows us to describe different aspects of the observational distribution via conditioning and marginalisation but does not allow us to express changes in the distribution through interventions. For example, $P(Y|T = 1)$ does not describe the interventional distribution of Y where all individuals were forcibly administered treatment but only describes the *subpopulation* of individuals who happened to get treated in the observational distribution.

The so-called do-operator was introduced to describe interventions and interventional distribution formally. As such, with the do-operator, we address the limitations of the standard statistical framework in capturing interventions and expressing causal queries.

For example, it allows us to express that we force all population members to receive treatment by writing $do(T = 1)$. Using the do-operator we can also express the distribution of Y given the intervention as $P(Y|do(T = 1))$ and the treatment effect, which compares the outcome probability under the intervention with treatment one to the outcome probability under the intervention with treatment zero, as $P(Y|do(T = 1)) - P(Y|do(T = 0))$.

In order to estimate interventional quantities, i.e., expressions involving a do-operator, we either need access to data from the respective interventional distribution or causal knowledge. Data from the interventional distribution can be collected by performing experiments; for example, so-called randomised controlled trials (RCTs) are the gold standard for inferring causal relationships in medicine. However, such experiments are often infeasible or unethical.

Thus, in many scenarios, we have to resort to observational data and causal knowledge. Over the course of this introduction to causal inference, we illustrate how causal knowledge can be used to estimate causal effects. However, before we can do that, we need a way to express causal knowledge. One type of causal model are so-called causal graphs.

Causal graphs and d-separation

In this thesis, we often use so-called causal graphs to visualise causal structure: We already encountered a causal graph in Figure 1.1 where I used it to visualise Simpson's paradox, in §1.4 I use a causal graph to visualise the model and data ecosystem in IML, and in several of our contributions we use causal graphs to illustrate what insight IML methods actually allow or to estimate causal effects.

In the causal graph $\mathcal{G}$, the nodes represent the modelled (endogenous) variables with index set D, and directed edges ($\rightarrow$) represent their causal relationships: The direct causes of a variable $j \in D$ – the endogenous variables that causally affect variable j, even if all other covariates are held constant via an intervention – are direct parents $pa(j)$ in the causal graph; Vice versa, the direct effects of a variable are direct children $ch(j)$ in the graph. This is also called the *Strict Causal Edge Assumption*. As a consequence, all causes are ascendants $asc(j)$ and effects are descendants $d(j)$.

Besides visualising the causal structure, causal graphs can be used to identify causal effects from observational data. Therefore, the causal structure must be linked to the dependence structure in the data. For such a link, further assumptions are required (beyond edges correctly indicating causal relationships):

- A (causal) graph is *acyclic* if no directed path starts and ends with the same node; Acyclicity ensures a range of desirable properties, among others that (given unchanged causal mechanisms) that the system induces a unique joint distribution [Bongers et al., 2021]. Directed acyclic graphs are also referred to as DAGs.

- The underlying probability distribution is *Markov* with respect to a DAG if each node j is independent of all its non-descendants $nd(j) := D\backslash d(j)$ given its parents, i.e., $X_j \perp X_{nd(j)}|X_{pa(j)}$ [Koller and Friedman, 2009].

If the causal graph is acyclic and Markov, we can read off conditional independencies in the data by inspecting the causal graph: Given the Markov property, d-separation in the graph $\mathcal{G}$ implies conditional independence in the data [Koller and Friedman, 2009], or more formally,

$$X_i \perp_d X_j|X_K \Rightarrow X_i \perp X_j|X_K.$$

To assess whether two nodes are d-separated by some set K, we check for each (undirected) path between the nodes, whether it is open or blocked by K. If all paths are blocked the nodes are d-separated, otherwise they are d-connected.

notation	meaning
D	index set for all endogenous variables
$pa(j)$; $ch(j)$	parents; children of node j
$asc(S)$; $nasc(S)$	descendants; complement
$d(S)$; $nd(S)$	descendants; nondescendants
$X_{pa(j)}$; $(X,Y)_{pa(j)}$	parents excluding; including Y

Table 1.2: Overview of our graph notation.

To assess whether a path is blocked, we regard each triplet on the path separately, if one triplet is blocked the whole path is blocked.
There are three possible types of triplets: chains ($X_l \rightarrow X_k \rightarrow X_m$), forks ($X_l \leftarrow X_k \rightarrow X_m$) and v-structures ($X_l \rightarrow X_k \leftarrow X_m$). Chains and forks are blocked if the middle node is in K, for v-structures it is the other way around [Koller and Friedman, 2009].
The three structures and the respective flow of association are visualised in Figure 1.3. Furthermore, we provide an overview of our graph notation in Table 1.2 and a visualisation thereof in Figure 1.2.

Observational identifiability

Before introducing causal graphs, we learned that estimating causal effects requires causal knowledge or experimentation. As follows, we briefly sketch how assessing the flow of association in causal graphs can help us to estimate causal effects from observational data. More specifically, we illustrate how to translate a causal expression (i.e., one involving a do-operator) into a statistical expression (i.e., one only involving the observational distribution).
Translating causal into statistical expressions is important in the context of IML, since many questions in IML are inherently about real-world causal effects, and access to interventional data is seldom available.
To illustrate the procedure, let us reconsider the mosquito example, illustrated in Figure 1.4b. By inspecting the graph, we see that there is an unblocked noncausal association path between mosquito prevalence M and tourism income I via the confounder season S. In this setting, we can use the so-called backdoor adjustment to estimate the effect anyway. The reason is that the noncausal association can be blocked by conditioning on the confounder season. More formally, the season variable satisfies the so-called backdoor criterion.

Definition 1 (Backdoor criterion)**.** *A set of variables C satisfies the backdoor criterion for treatment T and outcome Y, if*

1. *T blocks all backdoor paths, i.e. paths of association between T and Y that go via an incoming edge to T*
2. *C contains no descendants of the treatment T.*

Theorem 1 (Backdoor adjustment)**.** *If the backdoor criterion is fulfilled for the adjustment set C, the probability of Y given the intervention $do(T = t)$ can be*

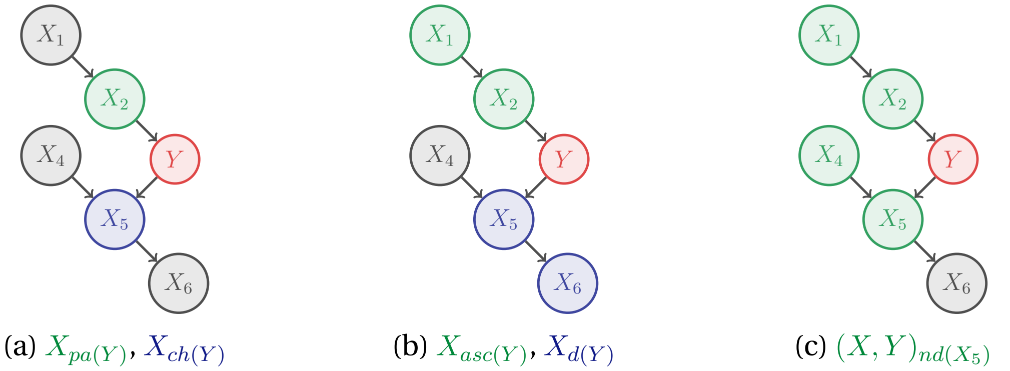


Figure 1.2: Visualisation of our graph notation.

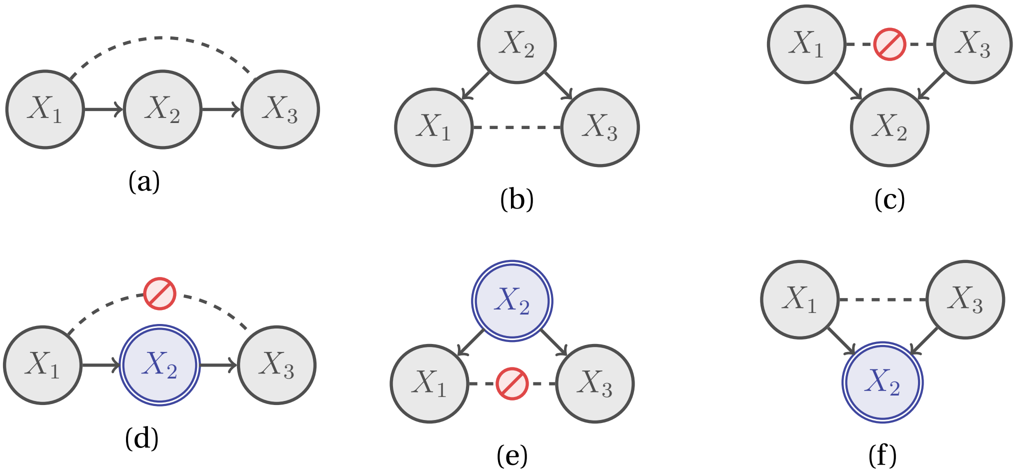
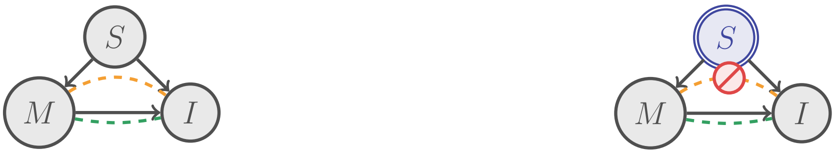


Figure 1.3: Three possible structures of paths with three elements X_1, X_2, X_3: A chain (left), a fork (centre), and a v-structure (right). The dotted lines visualise the association between nodes; the red forbidden sign indicates that the path of association is closed.
Top: If X_2 is not conditioned upon, X_1 and X_3 are not d-separated in the chain and fork structure, but are d-separated in the v-structure.
Bottom: When conditioning on X_2, the d-separations are flipped: In the chain and the fork structure X_1 and X_3 are d-separated, whereas in the v-structure X_1 and X_3 are not d-separated.



(a) Both causal and non-causal association between M and I.

(b) Only causal association between M and I after conditioning on S.

Figure 1.4: In this example, the goal is to assess the effect of the number of mosquitoes in Sweden M on the total income from tourism I. The size of the mosquito population is associated with the income from tourism via two paths: First, via the causal effect that mosquitoes have on tourist income (causal association). Second, via the confounder seasonality S (noncausal association), because seasonality affects both the size of the mosquito population and the number of tourists.
The noncausal flow of association can be blocked by conditioning on S (double lined). Thus, association and causation coincide within subgroups where the same day of the year $S = s$ was observed.

estimated using the adjustment formula

$$\sum_{c \in \mathcal{C}} P(Y|T = t, C = c)P(C = c).$$

In the mosquito example, the backdoor criterion is fulfilled for the adjustment set season S, since S blocks all backdoor paths between M and I. Thus – although the association between M and I is established via both causal and non-causal paths – in subgroups where all observations stem from the same season, only causal association remains. With the backdoor adjustment, we thus estimate the causal effects separately for each season and then aggregate the season-specific effects to yield the overall effect of mosquitoes on tourism income.
Of course, we cannot always find an adjustment set that satisfies the backdoor criterion and, therefore, cannot always estimate causal effects using the backdoor adjustment. To identify more causal effects, the do-calculus can be used. The do-calculus is a set of three implications that allow to translate interventional into associative statements (and also imply Theorem 1). Curiously, the do-calculus is *complete*, meaning that any causal effect that can be identified (from observational data) can be identified using the do-calculus.
Introducing the do-calculus goes beyond the scope of this thesis; We refer the interested reader to the literature [Pearl, 2009].

Counterfactuals and Pearl's ladder of causation

In the paragraphs on Simpson's paradox, we distinguished between association and causation. While associative questions concern dependencies in a given observational distribution, causal questions concern interventions and their effects. Furthermore, we introduced the do-operator to denote interventions formally and illustrated how interventional distributions can be identified with observational data.

As follows, we further differentiate between two distinct types of causal questions. In contrast to purely *interventional* queries, so-called *counterfactual* queries involve both a factual observation and a counterfactual world in which some intervention was performed. Such *counterfactual* questions are ubiquitous in human reasoning. For example, in medicine, we are often interested in understanding whether the outcome for a patient who received treatment A (factual world) would have been different if the patient had received treatment B (counterfactual world).

To express counterfactual queries, we need to introduce more notation. The reason is that counterfactual statements involve two states of the same variables, the factual pre-intervention state and the counterfactual post-intervention state. For example, to express the counterfactual probability of a favourable treatment outcome for a patient who had an unfavourable outcome before treatment, we need to distinguish between the pre-treatment and post-treatment outcomes. In this dissertation, we denote the pre-intervention state as Y^{pre} (the factual) and the post-intervention state as Y^{post} (the counterfactual). Counterfactual queries can be denoted as: $P(Y^{post}|do(T^{post} = t), Y^{pre} = y^{pre}, T^{pre} = t^{pre})$.

In general, access to the interventional distribution is not sufficient to answer counterfactual queries since the interventional distribution only involves the post-intervention states of the variables. Conditioning on the pre-intervention state is different from conditioning on the post-intervention state. For example, $P(Y|do(T = 1), Y = y^{pre})$ yields the post-intervention outcome distribution for a subgroup of individuals for whom the post-intervention outcome is y^{pre} – which is not what we are interested in. Because answering counterfactual queries requires access to distributions involving both pre- and post-intervention states, they require stronger causal knowledge than purely interventional queries [Holland, 1986, Pearl, 2009].

Pearl categorises the different types of noncausal and causal queries on what he calls the *ladder of causation* (Table 1.3). Queries that require stronger causal knowledge are placed higher on the ladder: Associative queries are placed on the first rung, interventional queries are placed on the second rung, and counterfactual queries on the third rung [Pearl and Mackenzie,

rung	name	concern
1	association	the observational distribution, e.g. prediction
2	intervention	interventional distributions, e.g. (conditional) treatment effects
3	counterfactuals	distributions involving both pre- and post-intervention states, e.g. what if one had been treated differently?

Table 1.3: Pearl's ladder of causation.

2018]. Higher-rung knowledge also allows answering lower-rung queries: If we have access to all counterfactual distributions, we also have access to all interventional distributions (e.g., by marginalising out pre-intervention variables); The observational distribution is the distribution with an empty intervention; as such access to all interventional distributions also gives access to the observational distribution.

Assuming unchanged circumstances, counterfactual statements can also be seen as *individualised* causal effect prediction, where the pre-intervention state of an individual is used to make the post-intervention prediction more accurate. For example, in our work on recourse (Paper I), we leverage the pre-intervention observation to tailor the effect estimate to the specific individual. Furthermore, we leverage both the pre- and post-intervention observation to decide whether somebody is qualified when the person reapplies after implementing recourse.

Structural Causal Models

In the preceding paragraphs, we distinguished between interventional and counterfactual queries and learned that counterfactual queries require stronger knowledge than interventional ones. To answer counterfactual queries, we need some way to transfer pre-intervention evidence into the post-intervention state. As we will demonstrate, so-called Structural Causal Models (SCMs) allow such a transfer.

While causal graphs only encode structural knowledge, SCMs also model the variables' functional relationships. Specifically, in SCMs, each variable is a function of its direct causes and its unobserved causal influences in the form of noise terms U. Modelling each variable as a function of its direct causes

captures the intuition of underlying (deterministic) causal mechanisms. The unobserved noise terms capture all stochasticity; Given their state, the observed variables are determined.
More formally, SCMs model the data generation process as a model $\mathcal{M} = \langle X, U, f \rangle$ that consists of the endogenous variables $X \in \mathcal{X}$, the mutually independent exogenous variables $U \in \mathcal{U}$, and the structural equations $f : \mathcal{U} \to \mathcal{X}$.[2] The structural equations are of the form:

$$X_j := f_j(X_{pa(j)}, U_j).$$

A model of the interventional distribution can be obtained by fixing the intervened-upon values to θ_I (e.g. by replacing the structural equation $f_I := \theta_I$).
The exogenous variables are the key to answering counterfactual queries. By reconstructing the exogenous variables from the evidence, for instance, using the inverse of the structural equations,[3] we can adapt the SCM to represent the specific individual and situation. The adapted SCM can then be used to reason about interventions while taking the pre-intervention evidence into account.
More formally, counterfactuals are computed in three steps: First, the evidence in the pre-intervention observation is used to reconstruct the exogenous variables U (*abduction*, i.e., learning $P(U_j|X = x^{pre})$). Second, the structural interventions corresponding to $do(a)$ are performed (*action*). Finally, we can sample from the counterfactual distribution $P(X^{post}|X = x^{pre}, do(a))$ using the abducted noise and the intervened-upon structural equations (*prediction*).
To conclude, SCMs allow answering questions on all three levels of Pearl's ladder of causation [Pearl, 2009, Pearl and Mackenzie, 2018].

Summary

In the preceding paragraphs, we distinguished between association and causation; Additionally, we differentiated between interventional and counterfactual causal queries. To formally denote causal queries, we introduced the do-operator and notation for factual and counterfactual states. Furthermore, we introduced two types of causal models that allow answering causal queries, namely causal graphs and structural causal models.

[2] The mutual independence ensures that no dependencies between the variables are induced by latent confounders

[3] If the structural equations are not invertible the distribution of possible states has to be learned (see later in the text).

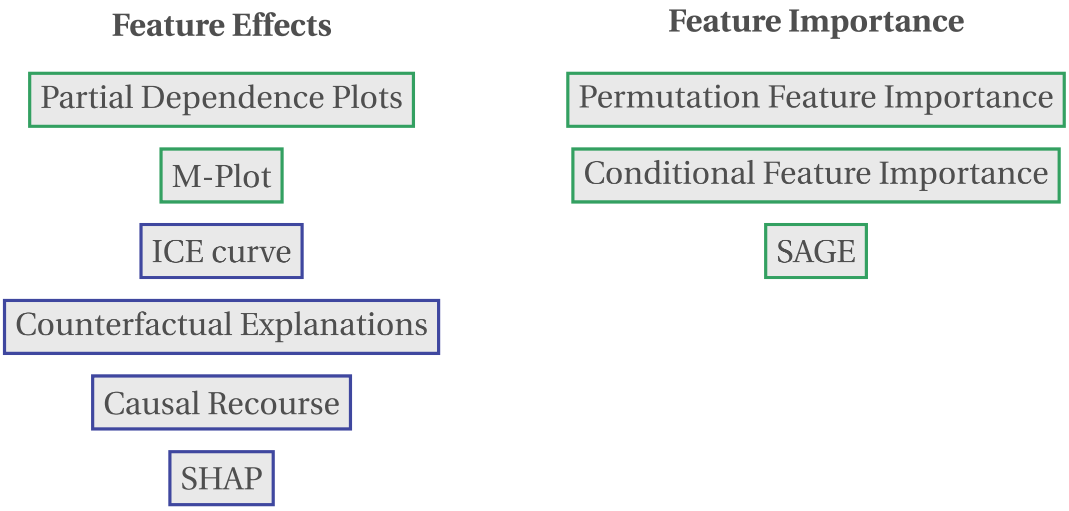


Figure 1.5: Overview of the model-agnostic post-hoc IML techniques covered in this introduction, structured according to commonly used attributes used to classify model-agnostic post-hoc interpretation methods: on the left, we see feature effect, on the right feature importance method. Local methods are coloured blue, global methods are coloured green. We will revisit this overview later in §1.5, where we categorise the methods in our own taxonomy.

1.2.3 Interpretable Machine Learning Methods

In this section, we introduce the interpretation techniques most relevant to our work, as well as several attributes commonly used to categorise them. For readers familiar with the IML literature, the section may be skipped. For a comprehensive overview of interpretable machine learning (IML), we refer the interested reader to the literature [Molnar, 2020, Holzinger et al., 2020].

Throughout this section, I will introduce the definitions of the following interpretation techniques: Individual Conditional Expectation (ICE) curves, Partial Dependence Plots (PDPs), Permutation Feature Importance (PFI), Conditional Feature Importance (CFI), Counterfactual Explanations (CEs), Causal Algorithmic Recourse (CR) and two Shapley-based methods called SHAP and SAGE (see Figure 1.5).

In this section, we formally introduce the methods but do not discuss their interpretation or usefulness. The interpretation of the methods is the concern of several of our contributions (§2).

Overview and Terminology

Albeit multiple works distinguish between terms such as *interpretable*, *explainable*, and *transparent*, all of these terms are overloaded with a range of different meanings [Doshi-Velez and Kim, 2017, Lipton, 2018, Guidotti et al., 2018, Weller, 2019, Krishnan, 2020]. In this dissertation, I use *explainable AI* (xAI) and *interpretable machine learning* (IML) interchangeably in their most abstract sense: as an umbrella term for methods that were associated with interpretability, explainability or transparency. Similarly, I interchangeably refer to the output of IML/xAI methods as *interpretations* or *explanations*.
Following the philosophy of science literature Woodward and Ross [2021], we refer to ground truth estimand that is to be explained as the *explanandum*. Furthermore, we will refer to the individual for whom the explanation is designed as *explainee*.
The field of IML subsumes methods that aim to generate *inherently interpretable models*, i.e. methods where the parameters can directly be made sense of and so-called *post-hoc* interpretation methods that aim to explain difficult-to-interpret models after the fact (i.e. without altering the model). Post-hoc interpretation techniques are further split into *model-specific* techniques, i.e., techniques that use model internals such as gradients, and *model-agnostic* techniques, that only require access to the model's prediction for queried feature values [Molnar, 2020].
In this thesis, we focus on the most flexible class of tools: model-agnostic post-hoc interpretation methods. Model-agnostic post-hoc methods can be further categorised into *local* and *global* methods. Local methods aim to explain only one specific prediction (i.e. one data point), whereas global methods aim to describe the model's behaviour over the whole domain [Molnar, 2020].
Furthermore, methods are often classified as *feature importance* or *feature effect* techniques. Although the terms are not used consistently, there is a rough consensus on their respective meanings: Feature effect methods typically refer to methods that explain the prediction $\hat{Y}$. In contrast, feature importance methods typically explain the model's performance, e.g. the risk R. Sometimes methods are referred to as importance methods because of the level of aggregation: For example, SHAP importance is not concerned with the model's performance but is also referred to as an importance measure, presumably because it globally explains the relevance of the variables with one value per feature.
Given the vagueness of existing terminology, I will later propose additional attributes which should be used to describe interpretation techniques (§1.4). For now, we use ”feature effect” to refer to methods that explain the prediction

and ”feature importance” to refer to methods that explain the performance.

Running Examples

For illustration, I apply the introduced methods to a running example.
I apply the methods on a random forest that was fitted on the UCI bike-sharing dataset [Fanaee-T, 2013]. The task is to predict bike rentals using features such as weather and seasonality. [4] For most methods I use my own implementation, for SHAP I rely on the homonymous python package.[5]

ICE curves and PDPs

Individual Conditional Expectation (ICE) curves. The ICE curve [Goldstein et al., 2015] is a local feature effects technique that, given an observation x, plots how the prediction $\hat{f}(x)$ changes if we replace the values for a set of features x_S. More formally, for an observation x, the ICE curve is defined as

$$ICE(x'_S; x) = \hat{f}(x'_S, x_C).$$

If the set S only contains one feature, the ICE curve can be plotted in a 2D line plot, where the x-axis corresponds to the value of x_S, and the y-axis to the respective model prediction. An exemplary plot is given in Figure 1.6 (thin grey lines). ICE plots only take one data point into account, and when we plot multiple curves for the different plots, the plot can get messy. Furthermore, if take more than one data point into account, a 2D visualisation is not possible.

[4] The random forest was fitted using `sklearn` and the default hyperparameter settings on 60% ($n = 439$) of the data. All methods are computed on the remaining 40% test data ($n = 292$).

[5] All code is available via GitHub (https://github.com/gcskoenig/diss-code.git). Whenever conditional sampling is employed, we use a default random forest with cross-entropy loss for categorical targets; For continuous targets, we assume that all dependencies between variables are captured in the conditional mean, and thus use a standard random forest regressor to predict the mean of the conditional distribution and resample the unexplained variance by permuting the residuals. For multivariate conditional distributions, we use a sequential sampling scheme [Bates et al., 2021, Blesch et al., 2023].

Partial Dependence Plots (PDPs). The partial dependence curve [Friedman, 2001] summarises the ICE curves over the whole domain by taking the expectation. More formally, partial dependence curves are defined as

$$\hat{f}_S(x_S) = E_{X_C}[\hat{f}(x_S, X_C))] = \int \hat{f}(x_S, X_C) dP(X_C).$$

Empirically the integral is estimated using Monte Carlo integration, i.e. by taking the average over the observed data points:

$$\hat{f}_S(x_S) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}(x_S, x_C^{(i)}).$$

The partial dependence function can be plotted for different values of x_S to yield the partial dependence plot. To enable visualisation, x_S is typically chosen to be univariate ($|S| = 1$). However, a 2D visualisation is possible as well. The thick black line in Figure 1.6 shows the PDP for the variable `atemp` in the bike-sharing dataset.

Notably, the marginal distribution of the remaining features, i.e. $P(X_C)$, is used when integrating out the remaining features. As such, dependencies between the feature of interest and the remaining features may be broken, and the model may be evaluated on unrealistic data points.

For example, suppose we have two temperature measurements from nearby locations. If we resample the first temperature measurement without taking its dependence on the second measurement into account, we create unrealistic feature combinations. E.g., it is implausible to have 20 degrees Celsius at Brandenburger Tor and minus 20 degrees Celsius at Bundestag and the model was not trained on such observations. Thus, it is debated whether the method should be used given dependent features [Apley and Zhu, 2020].

M-Plots. The so-called M-Plot has been suggested as an alternative [Apley and Zhu, 2020]. More specifically, M-plots resample the remaining variables from the conditional distribution $P(X_C|X_S = x_s)$ instead of the marginal distribution $P(X_C)$. Thus M-plots only evaluate the model within the observational distribution. More formally, they are defined as

$$\hat{f}_S(x_S) = E_{X_C}[\hat{f}(x_S, X_C))] = \int \hat{f}(x_S, X_C) dP(X_C|X_S = x_S).$$

Permutation Feature Importance and Conditional Feature Importance

Permutation Feature Importance (PFI). Permutation Feature Importance, [Breiman, 2001], sometimes called random forest feature importance, is one of the oldest feature importance techniques. It is based on a simple idea: To quantify the relevance of a variable for a machine learning model, PFI measures how much worse the model's performance is when we remove the variable. Since we cannot simply remove a variable from the model, PFI instead replaces the variable with a perturbed non-informative version.
More specifically, PFI for a dataset with n observations $x^{(1)}, \dots, x^{(n)}$ and some permutation π is defined as

$$PFI_j := \frac{1}{n}\sum_{i=1}^{n} L(\hat{f}(x_j^{\pi(i)}, x_{-j}^{(i)}), y^{(i)}) - \frac{1}{n}\sum_{i=1}^{n} L(\hat{f}(x^{(i)}), y)$$

PFI is a Monte Carlo estimate of

$$PFI_j := R(f(\tilde{X}_j, X_{-j}), Y) - R(f(X), Y)$$

with $\tilde{X}_j \sim P(X_j)$. An exemplary PFI plot can be found in Figure 1.7a.
PFI suffers from the same problem as PDPs: Since the perturbation is sampled from the marginal distribution, dependencies between features are ignored; Thus PFI evaluates the model outside the observational distribution (extrapolation).

Conditional Feature Importance (CFI). To avoid extrapolation, CFI was introduced [Strobl et al., 2008]. CFI samples the perturbation from the conditional distribution of the feature of interest given the remaining features $\tilde{X}_j^{-j} \sim P(X_j | X_{-j})$, such that dependencies between the covariates are preserved. CFI allows insight into the dependence of variables conditional on all remaining covariates [Strobl et al., 2008, Watson and Wright, 2021]. An exemplary plot can be found in Figure 1.7b.

Shapley-value-based methods

Shapley values are a game-theoretic concept that can be applied in interpretable machine learning as well. Before we get into the application in IML, let us quickly recapitulate what Shapley values are and what problem they solve.

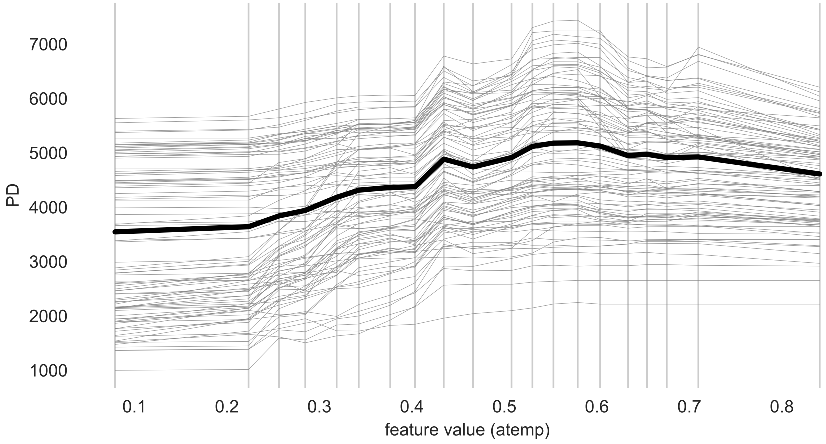
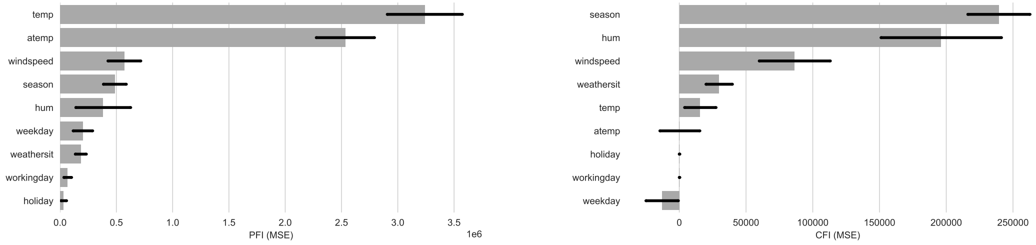


Figure 1.6: ICE curves and PDP for variable `atemp` in the bike sharing dataset.



(a) PFI

(b) CFI

Figure 1.7: Permutation Feature Importance and Conditional Feature Importance for the bike sharing predictor. While `atemp` and `temp` are the most important features for PFI, for CFI `hum` and `season` are more important.

Fair payoff in collaborative games. Suppose you take part in a quiz night and your group wins 100 coins. How should you fairly divide the payoff?
Equally dividing the money may be considered unfair, since players are not rewarded based on their individual performance. However, rewarding based on individual contributions is difficult, because the game is collaborative. That means that the surplus payoff that a player contributes depends on which players are already in the team, also referred to as the coalition. For example, the player may know an answer that another player in the coalition knows as well. Or the player may collaborate with another player to answer a question that none of them could alone.
Simple strategies fail to account for the nature of the game: If we quantify how much money each player would have won alone, we fail to account for collaborations. If we ask how much less the team had if the player did not join, we fail to attribute payoff for contributions that several team members share.

Shapley values. To attribute the payoff in such collaborative games fairly, Shapley values were proposed [Shapley et al., 1953]. Shapley values are the only attribution strategy that satisfies a range of fairness axioms:

- Efficiency: The Shapley values for all players add up to the overall payoff.
- Symmetry: Players who contribute equally to all possible coalitions receive the same payoff.
- Dummy: A feature that does not contribute to any coalition receives attribution zero.
- Additivity: For a game for which the total payoff is the sum of two subgames, the Shapley value of the overall game is the sum of the Shapley values of the subgames.

Given a function v that evaluates the payoff for sets of players S, the Shapley value is defined as

$$\phi_j(v) = \sum_{S \subseteq \{1,\ldots,d\} \setminus \{j\}} \frac{|S|!(p-|S|-1)!}{p!}(v(S \cup \{j\}) - v(S))$$

Intuitively, $v(S \cup \{j\}) - v(S)$ can be seen as the surplus contribution of player j over some coalition of teammates S. The Shapley value is the expected surplus contribution the player adds to randomly sampled coalitions.

Feature relevance quantification is a cooperative game. Like pub quizzes, the model's prediction (performance) can be seen as a collaborative effort where the features are the players, and the prediction (performance) is the payoff. Like a player's surplus contribution in a pub quiz, the surplus performance that a feature brings to the table depends on the coalition of features that it joins. Features may be dependent and thus may share information about the target. Furthermore, features may complement each other and ML models may thus rely on feature interactions.

A range of work argues that this collaborative nature should be considered when quantifying feature relevance and proposes to leverage Shapley values to do so [Datta et al., 2016, Lipovetsky and Conklin, 2001, Štrumbelj and Kononenko, 2014, Lundberg and Lee, 2017, Covert et al., 2020]. All aforementioned methods apply Shapley values, but rely on different value functions v. As follows, we introduce the value functions for two methods: SHAP [Lundberg and Lee, 2017] and SAGE [Covert et al., 2020].

SAGE. The goal of Shapley additive global importance (SAGE) values is to quantify the global relevance of the variables for the model's performance. As such, the payoff is the prediction performance, and the players are the variables. To quantify the prediction performance for a coalition S, the remaining features $-S$ are removed from the model via marginalisation.[6] This yields the restricted model f_S

$$f_S(x_S) = E[f(x)|X_S = x_S]. \tag{1.1}$$

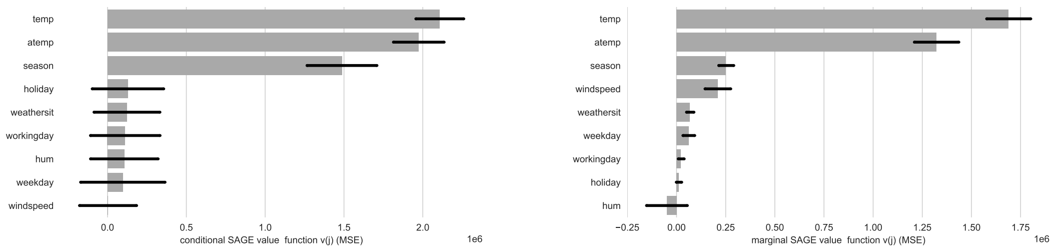
Based on the restricted function f_S the value function v is defined as the improvement in performance that the set S enables over the empty set $\emptyset$

$$v_f(S) = E[L(f_\emptyset, y)] - E[L(f_S(x_S), y)].$$

In practice, the value functions can be estimated using the respective empirical risks. Furthermore, given the difficulty of sampling from conditional distributions, the conditional expectation is commonly approximated with marginal sampling. Since sampling from the marginal alters the properties of the resulting SAGE values, we refer to the marginal-sampling-based variant as marginal SAGE, and the conditional-sampling-based variant as SAGE or conditional SAGE.

Exemplary plots for both marginal and conditional SAGE value functions and the respective SAGE values can be found in Figures 1.8 and 1.9. For more details about the theoretical properties of SAGE, confer [Covert et al., 2020].

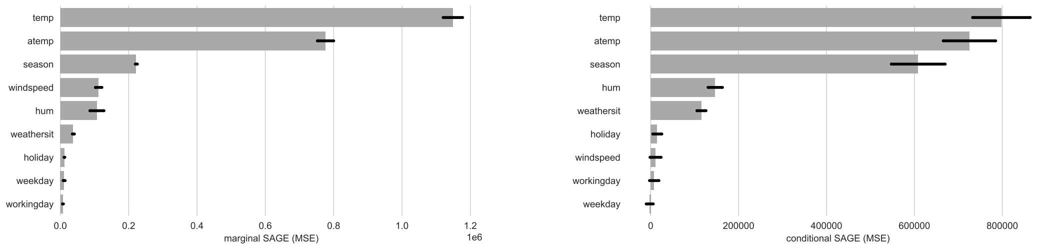
[6] In the original paper, the authors suggest using the conditional distribution $P(X_{-S}|X_S)$ for the marginalisation, but approximate the conditional using the marginal $P(X_{-S})$.



(a) marginal

(b) conditional

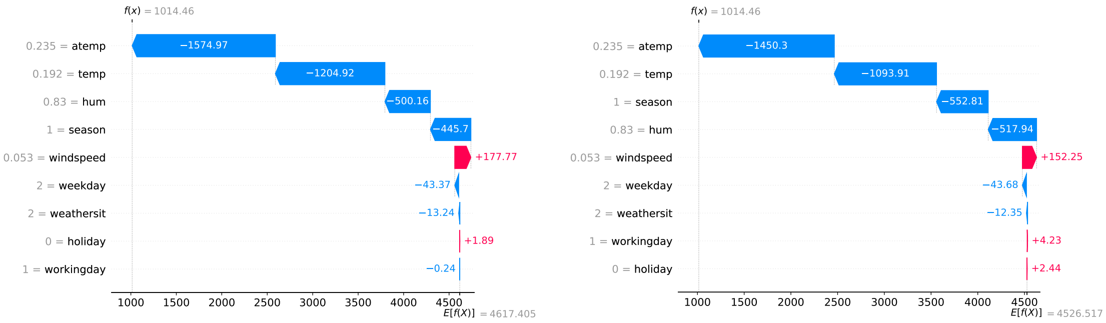
Figure 1.8: SAGE value functions.



(a) marginal

(b) conditional

Figure 1.9: SAGE values for the bike sharing dataset.



(a) Marginal-sampling-based SHAP values (estimated via permutation).

(b) Conditional-sampling-based SHAP values (estimated with TreeSHAP).

Figure 1.10: SHAP values for the bike sharing dataset. The bars indicate the SHAP values, which add up to the difference between the prediction for the data point and the mean prediction on the data set.

SHAP. Shapley additive explanations (SHAP) explain the prediction for a user-specified data point x; thus the feature values are the players, and the predicted value is the payoff. To quantify the payoff (the prediction) for a coalition S the restricted function f_S (Equation (1.1)) is used, i.e.,

$$v_f(S; x) = f_S(x_S).$$

Again, the conditional expectation of yielding the restricted function is commonly approximated with marginal sampling. We refer to the marginal-sampling-based variant as marginal SHAP and the conditional-sampling-based version as conditional SHAP. The output of both versions is visualised in Figure 1.10. For more details about the properties of SHAP values and their estimation, we refer to the literature [Lundberg and Lee, 2017, Janzing et al., 2020, Chen et al., 2020].

Contrastive Explanations

So-called contrastive explanations explain outcomes relative to one or more contrast cases [Lipton, 1990, Miller, 2019, 2021]. In interpretable machine learning, contrastive explanations are used to explain the outcome for a given data point by contrasting it with outcomes for alternative data points.

Counterfactual explanations (CEs). For counterfactual explanations [Wachter et al., 2017a], the model's prediction for a data point x is explained by contrasting it with data points that yield a different prediction. More formally, counterfactual explanations search for data points that are valid, meaning that they yield the desired prediction, and similar, meaning that the change δ is small.:

$$\text{argmin}_\delta \quad d(\delta + x, x) \qquad \text{s.t.} \qquad \hat{f}(x + \delta) = y'.$$

Here d is some distance function, x is the original datapoint, δ some change and y' the prediction outcome that we want to contrast against.

For example, when applied to the southern German credit example, for a randomly chosen rejected individual we get the following counterfactual explanation: ”When changing the ‘duration‘ to be 12 months shorter and set ‘other debtors‘ to guarantor, then the model's prediction is favourable.”

Wachter et al. [2017b] discuss three potential applications of counterfactual explanations: Understanding decisions, contesting (unethical) decisions, and altering future decisions (recourse).

Causal Recourse. In this thesis, we are especially interested in recourse. For recourse, the aim is to guide explainees to revert unfavourable decisions. It has been argued that it is important that recourse recommendations are *actionable* [Ustun et al., 2019], meaning that only changes that the user can realise are proposed, and *causal* [Karimi et al., 2020a], meaning that causal relationships between the features are taken into account.

Counterfactual explanations, as originally proposed, neither take actionability nor causal relationships between the features into account. Karimi et al. [2020a] thus modify the optimization problem to search for cost-minimal causal interventions that change the model's prediction. To estimate the causal effects of actions, they rely on causal models such as causal graphs and structural causal models [Karimi et al., 2021, 2020a,b]. For example, for SCMs with invertible structural equations, the optimisation problem is given by:

$$a \in \text{arg min}_{a\in\mathcal{F}} \quad cost(a; x^{pre}) \qquad \text{subject to} \qquad h(x^{post,do(a)}) \geq 0.5,$$

where $cost(a, x^{pre})$ measures the cost of action a for an individual with pre-recourse characteristics x^{pre}, and where $x^{post,do(a)}$ is the corresponding post-recourse state for action a.

1.3 Challenges for Interpretable Machine Learning

IML is a vibrant research field, and the methods are widely applied in practice [Fellous et al., 2019, Deeks, 2019, Gade et al., 2019, Gordon et al., 2019, Danilevsky et al., 2020, Jiménez-Luna et al., 2020, Tosun et al., 2020, Das et al., 2021, Tantithamthavorn and Jiarpakdee, 2021, Sharma et al., 2022, Yang, 2022, Khosravi et al., 2022, Gevaert, 2022, Machlev et al., 2022, Fiok et al., 2022]. At the same time, the field faces fundamental criticism.

As follows, I summarise the criticisms of the field that, in my view, are the most pressing: First, we argue that interpretability conflates several incompatible goals that must be disentangled; Only given a fixed goal the target estimand (*explanandum*) can be derived (§1.3.1). Second, to establish a link between *explanation* and *explanandum*, we need interpretation rules that make clear what aspects of model and data a method provides insight to (§1.3.2). Thirdly, many IML methods are difficult to estimate (§1.3.3). The challenges are illustrated in Figure 1.11.

In §3.1 we discuss how our contributions help to tackle them.

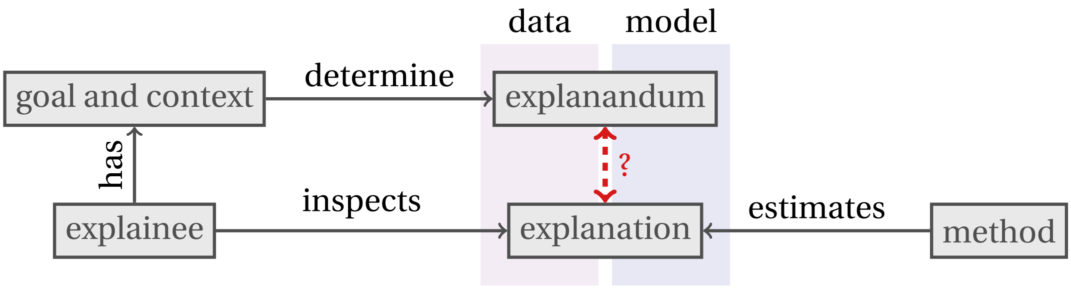


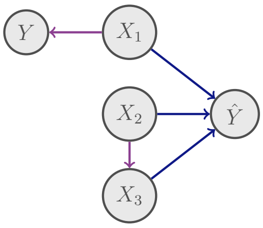
Figure 1.11: An illustration of the explanatory gap between interpretation goal and method. In order to establish a link, we have to understand which aspects of model and data are relevant for a given interpretation goal (*explanandum*), and which aspects the IML method describes (*explanation*).

1.3.1 Challenge I: Unclear explanandum

A common criticism of IML is that the goal of the field is to make opaque machine learning systems ”interpretable” but that there is no agreement on what that is supposed to mean [Lipton, 2018, Krishnan, 2020, Freiesleben and König, 2023]. As such, claims about “interpretability” are not meaningful without further clarification, and aiming for “interpretability” is an ill-posed problem.

To better understand the origin of the issue, we take a step back and reconsider why we care about “interpretability” in the first place: As Doshi-Velez and Kim [2017] argue, we ask for explanations since there are important requirements towards machine learning systems that cannot be evaluated with test set performance. These requirements include (among others) the compliance of the system's mechanism with ethical standards (fairness), the robustness of the system to domain shifts (robustness), the ability to learn from the model about the data-generating mechanism (inference), and the ability to explain algorithmic decisions in a way that allows affected individuals to revert unfavourable decisions (recourse). The hope associated with interpretability is that if we were to ”understand” the ML system, we would be able to assess whether a system is fair and robust, learn about the data-generating process or restore the agency of individuals affected by unfavourable decisions.

Problematically, these requirements conflict. Let us illustrate this with an example illustrated in Figure 1.12. The model relies on the two uninformative features x_2 and x_3 that cancel out in the observational distribution (meaning that within the observational distribution $\hat{f}(x) \approx x_1$). If we interpret the model to find which variables are most predictive of y (inference), then it is



(a) DAG

$$
\begin{array}{ll}
X_1 := \epsilon_1,\, X_2 := \epsilon_2 & \epsilon_1, \epsilon_2 \sim N(0,1) \\
X_3 := X_2 + \epsilon_3 & \epsilon_3 \sim N(0, 0.01) \\
Y := X_1 + \epsilon_Y & \epsilon_Y \sim N(0, 0.1)
\end{array}
$$

$$\hat{y} = \hat{f}(x) := x_1 + cx_2 - cx_3$$

(b) SCM

Figure 1.12: In this example, the model relies on two features x_2, x_3 that are independent of Y and that cancel each other out in the observational distribution.

irrelevant to know that the model relies on x_2 and x_3 because we only care about the data, and the variables have no relation with y. More generally, if we want to assess the model's behaviour in the observational distribution, x_2 and x_3 are irrelevant. However, if we interpret to model to assess its robustness to distribution shifts, we want to know that the model's mechanism includes the term $cx_2 - cx_3$ and, thus, that its performance breaks down once the correlation between the two variables is broken.

More generally, the questions that individuals may have when interpreting machine learning models are so heterogeneous that no single method can answer them all concisely. Instead, as Lipton [2018] puts it in his seminal paper,

> *[...] interpretability is not a monolithic concept but several distinct ideas that must be disentangled before any progress can be made.*

1.3.2 Challenge II: Misinterpretation of IML Methods

Not only the ML model but also the IML methods are subject to interpretation: Given the diversity of contexts and motivations to seek explanations, the IML methods are diverse in their actual meanings. To choose an IML method that suits the explanation context and goal, the explainee must be aware of what insights the different method can and cannot provide.

In a recent study Krishna et al. [2022] demonstrate that practitioners from research and industry lack orientation: When presented with the outputs of several popular IML methods, they find that the methods commonly disagree; For example, because the highest ranked features differ, the overall relative ordering differs, or even the signs of the values differ. [7] When asked to choose

[7] This result is in agreement with Figures 1.6-1.10, where we plot several IML methods on the bike sharing dataset and observe that they yield different results

one of the methods to resolve the disagreement, participants predominantly rely on superficial criteria such as publication year or on whether the method's output matches their prior intuition.

The study's results are sobering: If explainees do not understand what the explanations actually mean and instead base their choice on superficial criteria, the conclusions they draw from the explanation may not be grounded in reality. Even worse: If explanations are chosen based on whether they match the explainee's prior intuition, they degrade to (meaningless) justifications.

To prevent such misinterpretation, we need more formal work clarifying what insight the different IML methods provide. Furthermore, the resulting insights and misconceptions must be communicated to a wider audience.

Beyond unclarity about what aspect of model and data a method targets, the IML estimate is subject to various uncertainties that can significantly alter the explanations; These uncertainties must be quantified and communicated to the explainee [Watson, 2022].

1.3.3 Challenge III: Estimation of Conditional-Sampling-Based Methods

IML methods typically quantify the relevance or effect of features by quantifying how perturbing the feature affects the object of interest (e.g. the prediction). A range of methods thereby rely on marginal sampling, e.g. by permuting the values for the variable; This has the disadvantage that the dependencies with the remaining variables are ignored, such that observations in unseen or unrealistic regions are created. A range of work therefore argues in favour of conditional-sampling-based perturbations that take the dependencies between features into account [Strobl et al., 2008, Hooker and Mentch, 2019, Frye et al., 2020, Chen et al., 2020, Freiesleben et al., 2022, Freiesleben and König, 2023].

However, conditional sampling is difficult and computationally expensive; for many methods, conditional-sampling-based implementations are not available. Thus, the methods are less convenient, and methods are ”approximated” with marginal sampling instead (e.g. Lundberg and Lee [2017], Covert et al. [2020]).

1.4 Nine Perspectives on Model and Data

The limits of my language mean the limits of my world.
– Ludwig Wittgenstein

In the preceding section, we argued that IML faces fundamental challenges; the unclear terminology in the field compounds these problems. For example, IML methods are often said to quantify a feature's relevance but without clarifying what it means to be relevant. If we fail to articulate the meaning of the method, how should practitioners be able to make the right conclusions?

Throughout this section, we will refine our terminology. More specifically, we leverage a causal viewpoint to distinguish between nine different perspectives on model and data. To illustrate the different perspectives, we look at exemplary questions relevant in IML and categorise them according to what perspective they concern; Thereby, we demonstrate the usefulness of the taxonomy to articulate the *explanandum* more clearly (Challenge I). Furthermore, we categorise the IML methods introduced in §1.2.3 within the taxonomy, thereby tackling the misinterpretation of IML methods (Challenge II). In summary, the taxonomy serves as a connecting link between *explanation* and *explanandum*.[8]

For now, let us understand what these nine different aspects are – with a focus on understanding that they are indeed distinct. We structure the taxonomy using two questions: First, what object is described? The prediction $\hat{Y}$, the underlying target Y or their relationship R (as captured by the risk)? Secondly, on what level are we trying to understand the object? In terms of model-level intervention, data-level interventions or in terms of association?

We illustrate the distinction at the example of disease diagnosis, where causes C and symptoms S are used to diagnose the disease state Y (Figure 1.13).

1.4.1 What Object is Described?

Except for special cases where we can perfectly reconstruct Y from the features, prediction and target take different values. Moreover, even if we can predict perfectly on test data, Y and $\hat{Y}$ may take different causal roles. For example, disease diagnosis models (as visualised in Figure 1.13) may rely on disease symptoms for their prediction, such that treating the symptoms

[8] It must be emphasised that connecting explanation and explanandum in the taxonomy is necessary, but not sufficient to establish an explanatory link.

causally affects the prediction $\hat{Y}$; However, the symptoms are by definition not causal for the disease, and thus treating them does not affect Y.
The relationship between Y and $\hat{Y}$, e.g. quantified as the risk $R(Y,\hat{Y})$, differs from both Y and $\hat{Y}$; It takes different values and plays a different causal role.

1.4.2 On What Level is the Object Described?

In the preceding paragraph, we differentiated between the three objects of interest: The prediction $\hat{Y}$, the underlying target Y and their relationship. In the context of IML, we typically try to understand these objects in terms of their relationship with the features.[9]
As follows, we differentiate between three levels on which the relationship between features and the object of interest can be described. First, we distinguish between causation and association (using the do-operator); Secondly, we distinguish between model-level causation and data-level causation.

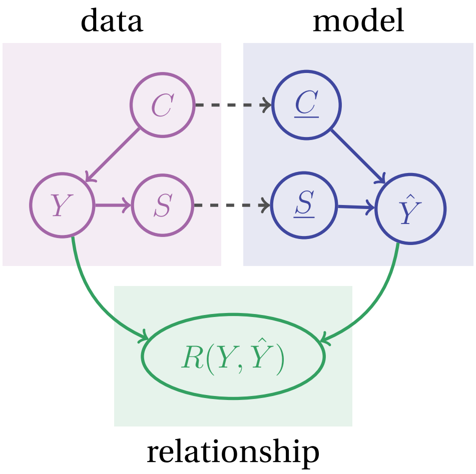


Figure 1.13: Visualisation of our distinction between different aspects of model and data, at the example of a model that uses causes C and symptoms S to diagnose a disease state Y. On the left side, we see the real-world variables and their causal relationships, and on the right side, the respective model counterparts (the model inputs $\underline{C}$ and $\underline{S}$ and the model's prediction $\hat{Y}$). The relationship between prediction and target is captured by the risk R.

[9] For example, feature effect methods describe the relationship between the feature and the prediction, and feature importance methods quantify the relevance of features for the model's performance.

Distinction between causation and association. Association does not imply causation – this statement is also true in the context of IML. In the disease diagnosis example (Figure 1.13), the symptom state S is associated with the disease Y, but the symptoms S are not causal for the disease Y.
We use the do-operator (introduced in §1.2.2) to denote the difference between intervention (causation) and observation (association): When we observe symptom state s, we write $S = S$; if we intervene on the symptoms to take value s, we write $do(S = s)$.

Distinction between model-level and data-level causation. When referring to causality in interpretable machine learning, we must further distinguish between *model-level* interventions and *data-level* interventions. With data-level interventions, we mean interventions in the real world. With model-level interventions, I mean interventions on the model inputs (by plugging different values into the predictor).
These two notions of causal effects are fundamentally different: For data-level causation, the causal dependencies between variables in the real world must be considered; for model-level causation, they are deliberately ignored: In the disease diagnosis example (Figure 1.13), real-world interventions on the causes C affect the prediction $\hat{Y}$ via two paths: directly via the changed model input, and indirectly via the effect on the symptoms S. In contrast, intervening on the corresponding model input does not affect the symptoms and thus yields a different effect.
To formally distinguish between model-level and data-level interventions, we follow Janzing et al. [2020] and introduce a separate variable for the model input, distinguished by an underline (e.g., $\underline{C}$ for the disease causes model input). We regard the model input as a perfect copy of the respective real-world state, i.e. define the respective structural equation as $\underline{C} := C$. An intervention on the model input is denoted as $do(\underline{C} = c)$.

So to conclude, we distinguished between three objects (the prediction $\hat{Y}$, the underlying target Y, and their relationship) and three levels of description (association, model-level causation and data-level causation). In combination, we yield nine aspects of model and data.

1.4.3 Heterogenous Explananda

To illustrate the different aspects of model and data, we collect exemplary questions that may have when interpreting a disease diagnosis model and categorise them in the taxonomy.
For example, in order to be able to simulate the model's behaviour, we may be interested in understanding the model's mechanism, i.e., how model-level interventions affect the prediction. In contrast, to identify predictive biomarkers, we are interested in associations between the features and the prediction target. Or, as someone being diagnosed with a disease, we may be interested in understanding what we can do to get healthy, which requires estimating the effect of data-level interventions on the underlying target. More examples can be found in Table 1.4.[10]

	$\hat{Y}$	R	Y
$do(\underline{X} = x')$	Does the model's mechanism rely on gender?	Do measurement errors affect the performance?	-
$X = x'$	Is the diagnosis correlated with gender?	Is the diagnosis more accurate for men?	Which biomarkers are predictive of Y ?
$do(X = x')$	What can I do to appear healthy?	Does the model work in different hospitals?	What can I do to get healthy?

Table 1.4: Exemplary queries for different aspects of model and data.

We observe that the questions concern distinct aspects of model and data, implying that distinct methods are required to address them. Thus, the categorisation confirms the argumentation in Section 1.3.1: Each question corresponds to a different notion of what it means for a ML system to be "interpretable" and any "one-fits-all" definition of interpretability must conflate several incompatible requirements.

[10] I do not give an example for a question that concerns the effect of model-level interventions on the underlying target, since in our simplified model the prediction does not affect the real-world state. However, such feedback loops are conceivable [Perdomo et al., 2020].

1.4.4 Heterogeneous Explanations

Furthermore, we categorise the IML methods that were introduced regarding the aspect of model and data that they compute in Table 1.5.[11]

Marginal-sampling-based methods measure the effects of plugging perturbed versions of the variables into the model while ignoring any dependencies between the variables. As such, marginal-sampling-based methods measure the effects of model-level interventions. While feature effect methods such as PDPs concern the prediction, feature importance methods such as PFI concern the model's performance.

Conditional-sampling-based methods measure the objects of interest conditional on observing (a subset) of the variables. For instance, M-Plots describe the expected prediction conditional on observing the feature of interest, and SAGE value functions describe the model's performance conditional on observing a set of features.

Only the Causal Recourse (CR) method is concerned with interventions on the data level.

	$\hat{Y}$	R	Y
$do(\underline{X} = x')$	ICE plots and PDPs marginal SHAP CEs	PFI marginal SAGE	-
$X = x'$	M-Plots conditional SHAP	CFI conditional SAGE	-
$do(X = x')$	CR	-	-

Table 1.5: The interpretation techniques introduced in §1.2.3, categorised regarding what aspect of model and data they portray.

When confronted with a choice between different IML methods for a given task, the taxonomy helps to narrow down the candidate pool significantly.

[11] Tables 1.4 and 1.5 illustrate that there is a gap between the aspects of the model and data that may be of interest in IML and what the methods actually compute. With the contributions in this thesis, we aim to close this explanatory gap in two ways: By clarifying target estimands (understanding what aspect of model and data we care about), and by studying what insight into model and data actually allow.

However, it may be noted that the taxonomy has its limitations: First of all the nine categories tell us from which perspective model and data are approached, but does not fully determine what we inspect. For instance, we are agnostic to how interactions are handled or to whether local or global explanations are generated. Thus clarifying the targeted perspective is necessary but not sufficient for establishing an explanatory link. Second, the taxonomy is explaining via the features and thus does not accommodate all IML methods. For instance, models may be explained via model internals or via specific data points as well. And Third, depending on the assumptions that we can make about model and data, there may be significant overlap between the categories; Thus when method and goal concern different perspectives on model and data, it depends on model and data whether an explanatory link can be established anyway.

1.4.5 Overlap Between the Nine Perspectives

Although the nine aspects are distinct, there can be significant overlap. If prediction models are Bayes optimal, they can be seen as models of properties of $P(Y|X)$, establishing a relation between Y and $\hat{Y}$ (§1.2.1). And the do-calculus relates (conditional) probability statements with causal statements (§1.2.2).
For instance, although the partial dependence plot (PDP) concerns the effect of model-level interventions on the prediction $\hat{Y}$, it can also be used to estimate the effects of data-level interventions on the underlying target Y [Zhao and Hastie, 2021]. More specifically, the PDP and the adjustment formula coincide, meaning that if the set of remaining variables satisfies the backdoor criterion (and the model is accurate in the regions in which the PDP evaluates it), the PDP visualises a real-world causal effect.

1.4.6 The Nine Perspectives and Our Contributions

The taxonomy will prove helpful over the course of this thesis and is the basis of several of our contributions. It helped us to discuss the requirements and the explanandum in the context of *recourse* (Paper I) and *scientific inference* (Papers II and III) and served as a grid to assess what insight a general class of feature importance method allows (Paper IV).

1.5 Overview of Our Contributions

The thesis includes seven papers, which contribute to tackling the challenges postulated in Section 1.3.

- To tackle unclear interpretation goals and target estimands (Challenge I), we investigate two interpretation contexts in more detail: recourse (Paper I) and scientific inference (Papers II-III). In each setting, starting from the interpretation goal we motivate the target estimands and discuss which interpretation methods are suitable.

- Papers I-V are concerned with preventing misinterpretation (Challenge II). In Paper I we clarify that existing recourse methods may fail to lead to improvement, which we argue to be a vital requirement for recourse. In Paper II we establish which methods can or cannot be used to gain insight into the data-generating process, and in Paper III we propose methods to quantify the uncertainties involved when linking explanations to properties of the DGP. In Paper IV we generalise PFI and CFI to a general class of feature importance algorithms and derive interpretation rules for each of its members. In Paper V, we communicate general interpretation pitfalls, such as unjustified causal interpretation, to a broader audience.

- In Papers VI-VII we propose methods to estimate conditional-sampling-based methods efficiently. In Paper VI, we use causal structure learning (CSL) to greedily identify the dependence structure in the data which can be exploited to make the estimation of SAGE values more efficient. In Paper VII we rely on tree-based models to learn a partitioning of the feature space that renders the covariates independent (conditional on the partitioning), which can be exploited to sample from conditional distributions by permuting observations within the partitions.

As follows, I shortly summarise our contributions. A visual overview of the articles and their relation to the three challenges is given in Figure 1.14.

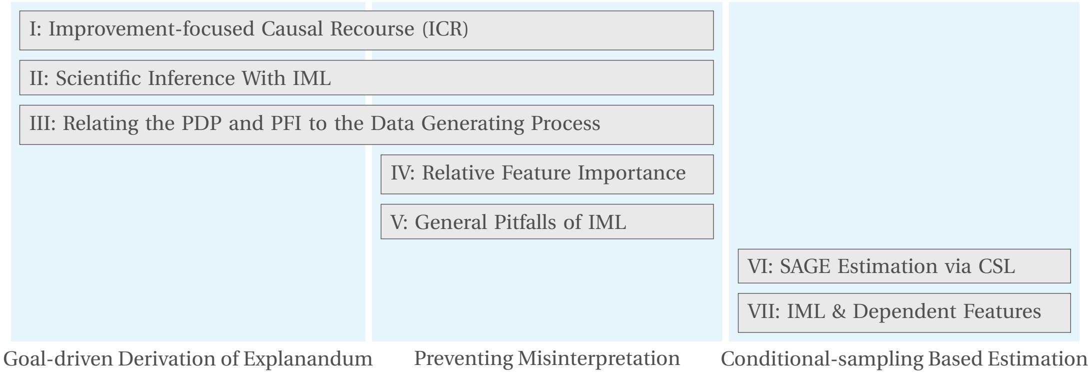

I: Improvement-focused Causal Recourse (ICR)
II: Scientific Inference With IML
III: Relating the PDP and PFI to the Data Generating Process
IV: Relative Feature Importance
V: General Pitfalls of IML
VI: SAGE Estimation via CSL
VII: IML & Dependent Features
Goal-driven Derivation of Explanandum
Preventing Misinterpretation
Conditional-sampling Based Estimation

Figure 1.14: Overview of the seven papers in this thesis and their relation to the three challenges postulated in Section 1.3.

Paper I: Improvement-Focused Causal Recourse (Section 2.1). An important application of IML is providing so-called recourse recommendations: Suppose you applied for a loan, and an ML model is used to assess your qualification. Suppose, furthermore, that the model rejects you. You feel powerless since you do not know how the decision was made or how you can change it. So-called recourse recommendations aim to restore your agency by telling you what you can do to revert this unfavourable decision. A recourse recommendation could be of the form: ”If you reduce your credit card risk by 1000€, you will get the loan.”
A range of recourse techniques have been proposed: counterfactual explanations [Wachter et al., 2017b], counterfactuals with actionability constraint [Ustun et al., 2019], and the causal recourse framework [Karimi et al., 2020a]. We demonstrate that all existing methods suffer from a fundamental problem: the methods target *acceptance*, meaning that the prediction $\hat{Y}$ is reverted but may fail to guide towards *improvement*, meaning that they may fail to revert the underlying target Y. In other words: Existing recourse techniques may recommend to *game* the predictor.
In our paper on *Improvement-focused Causal Recourse (ICR)*, we argue that improvement is a fundamental requirement for recourse since it is desirable for model authority, explainee and society: Model authorities have no incentive to recommend tricking their decision system. For the explainee, gaming may have short-term benefits and be morally justified if the predictor is unethical; However, gaming recommendations may mislead explainees who are actually interested in improving, which in general, is a more robust long-term strategy. Furthermore, explainees are members of society; For society, it is important that recourse also leads to improvement since gaming damages collective risk systems. For instance, if loans are given to individuals

whose default rate is underestimated, the financial system may be seriously damaged.

Thus, we propose an improvement-focused recourse technique. Depending on the level of causal knowledge, more or less accurate recommendations can be made. If a Structural Causal Model (SCM) is available, we can leverage structural counterfactuals for individualised effect prediction. If only the causal graph is available, we can estimate conditional average treatment effects instead. We refer to the two settings as the *individualised* and the *subpopulation-based* setting.

For both settings, we derive acceptance guarantees from improvement guarantees. In the *subpopulation-based* setting, where the improvement rate is estimated using a causal graph, we show acceptance bounds for Bayes-optimal observational predictors. In the *individualised* setting, where the effect estimate is individualised using the SCM, we propose to leverage the SCM for individualised post-recourse prediction, for which we derive acceptance bounds.

In various synthetic and semi-synthetic settings, we empirically demonstrate that ICR reliably guides towards improvement and acceptance while being more robust to refits of the model than counterfactual explanations or causal recourse.

	$\hat{Y}$	R	Y
$do(\underline{X} = x')$	Counterfactual Explanations (CEs)	-	-
$X = x'$	-	-	-
$do(X = x')$	Causal Recourse (CR)	-	Improvement-focused Causal Recourse (ICR)

Table 1.6: While both counterfactual explanations and causal recourse target *acceptance* (reversing $\hat{Y}$), we argue that recourse should lead to *improvement* (reversing Y).

Papers II and III: Scientific Inference with Interpretable Machine Learning (Sections 2.2 and 2.3). Machine learning models are increasingly deployed in science. The reason for their adoption is their ability to model complex dependencies, allowing for more accurate prediction than classical statistical models.

Beyond accurate prediction [Luk, 2017, Douglas, 2009], science is concerned with learning about the data-generating process (DGP) [Salmon, 1979, Longino, 2018, Shmueli et al., 2010]. Problematically, machine learning models are often too complex to understand, and it is unclear whether individual model elements can be linked to properties of the DGP. Thus, scientists struggle to learn about the DGP using ML models.

Interpretable Machine Learning (IML) is concerned with tackling the opacity of ML models by describing elements of the models or properties of model and data. Since model elements may not represent DGP properties, and since IML methods are developed with a wide range of different contexts and goals in mind, it is often unclear what insight the methods provide (Challenge II, Figure 1.15), and therefore whether the methods enable scientific inference. Nevertheless, IML methods are increasingly deployed to infer the ”relevance” of features in scientific applications [Fellous et al., 2019, Gade et al., 2019, Gordon et al., 2019, Danilevsky et al., 2020, Jiménez-Luna et al., 2020, Tosun et al., 2020, Das et al., 2021, Tantithamthavorn and Jiarpakdee, 2021, Sharma et al., 2022, Yang, 2022, Khosravi et al., 2022, Gevaert, 2022, Machlev et al., 2022, Fiok et al., 2022], and it is often implied that the explanations actually reflect properties of the DGP.

We tackle the challenge in Papers II and III.

In Paper II, *Scientific Inference with Interpretable Machine Learning*, we show that IML methods can indeed describe interesting properties of the DGP, more specifically, of the underlying joint probability distribution. The reason is that – although individual model elements may not represent DGP properties – the model *as a whole* represents aspects of the underlying distribution. For example, regression models that minimize the mean squared error represent the conditional expectation of the prediction target given the covariates. We call this *holistic representationality*.

Nevertheless, since IML addresses several conflicting goals, many existing methods do not enable insight into the DGP. We identify the properties that IML methods must fulfil to be suitable for scientific inference about the joint distribution of the variables. Based on these insights, we propose a guide for scientific inference with IML that comprises 5 steps: 1) Formalising the scientific question, 2) establishing identifiability, 3) designing an IML property descriptor, 4) estimating the descriptor and 5) quantifying uncertainty.

Quantifying uncertainty is the focus of Paper III, *Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process.* More specifically, we formalize *permutation feature importance* and *partial dependence plots* as statistical estimators of properties of the DGP and show that the estimates deviate from the ground truth not only due to statistical biases but also due to learner variance and Monte Carlo approximation errors. To avoid misinterpretation, we propose learner versions of PD and PFI that are based on model refits, as well as variance and confidence interval estimators that account for the involved uncertainties.

In both papers, we focus on the supervised learning paradigm and inference about associations in the data. We briefly discuss other forms of inference, especially causal inference, but leave a more detailed assessment for future work.

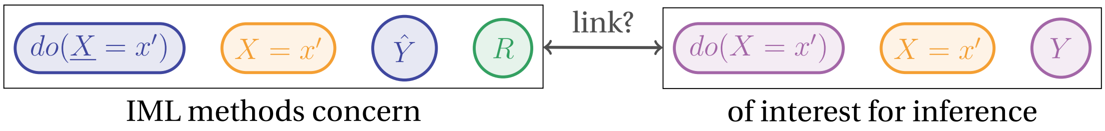


Figure 1.15: While IML methods are typically concerned with the effect of model-level interventions on prediction and performance, inference is concerned with associations and causal relationships in the data-generating process.

Paper IV: Interpretation Rules for Feature Importance (Section 2.4). Feature importance methods quantify the relevance of features by measuring the impact of feature perturbations, i.e. interventions on the model level, on the model performance. Although the effect of feature perturbations on the model performance is easy to compute, it is unclear what conclusions can be drawn. In our paper on *Relative Feature Importance* we thus investigate the insight that nonzero feature importance provides about the model's mechanism and the dependencies in the data (Figure 1.16).

Depending on the concrete feature importance method, different model-level interventions are performed. For example, for Permutation Feature Importance (PFI), the feature is resampled from its marginal distribution, and for Conditional Feature Importance (CFI), the feature importance is resampled from the conditional distribution given all the remaining features. These methods yield different results, requiring different interpretations.

For our assessment, we thus generalise PFI and CFI to a general class of feature importance methods, which we call *Relative Feature Importance (RFI)*, and derive general interpretation rules that can be applied to each class member.

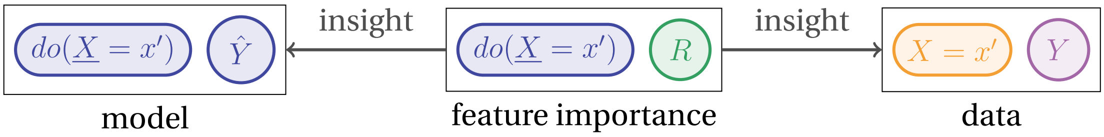


Figure 1.16: In Paper IV, we study what insight into model and data we can gain by inspecting Relative Feature Importance (RFI), a class of feature importance methods.

Paper V: Raising Awareness for Interpretation Pitfalls (Section 2.5). So far, we have tackled the misinterpretation of IML (Challenge II) by establishing relationships between feature importance algorithms and different aspects of model and data (Paper IV) or by proposing a guide on how to choose the interpretation technique (Paper II).

In Paper V, *General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models*, we approach the problem *ex negativo*, meaning that we aid practitioners by exposing what *not* to do. More specifically, we identify common mistakes that may be made when choosing and interpreting IML methods. The pitfalls include: Assuming using interpretation methods for unsuitable purposes, interpreting models that do not generalise well, ignoring feature dependence and feature interactions, ignoring model and approximation uncertainty, as well as unjustified causal interpretation. We illustrate each pitfall on an example, offer solution strategies and discuss open issues.

Paper VI: Efficient SAGE value estimation via Causal Structure Learning (Section 2.6). SAGE values are a theoretically appealing feature importance method: They can be linked to properties of the data-generating process, such as conditional mutual information. Furthermore, they fairly attribute importance according to the Shapley fairness axioms; That means that they provide insight into the relevance of features beyond dependence on no or all features.

One downside of SAGE values is their computational inefficiency. The exact computation of SAGE values requires the evaluation of an exponential number of so-called surplus function evaluations. This is particularly expensive because the evaluation of the surplus functions requires sampling from potentially multivariate conditional distributions.

In this work, we aim to reduce the computational burden of SAGE value estimation by exploiting the observation that conditional independence in the data implies that the respective surplus contribution evaluates to zero, such that their computation can be skipped. More specifically, we leverage causal structure learning – greedy algorithms that learn the (conditional) independence structure in the data – to factorise the distribution.

As we demonstrate empirically, this is more efficient since the one-time computation effort required to learn the dependence structure is negligible in comparison to the expense of the many saved surplus contribution evaluations. Furthermore, since the false discovery rate for the employed CSL algorithms is close to zero, we yield unbiased approximations.

Paper VII: Conditional Sampling Based Feature Importance and Feature Effects (Section 2.7). Conditional Feature Importance (CFI) is theoretically appealing since it allows insight into the conditional (in)dependencies in the data; At the same time, it requires sampling from the conditional distribution of the feature of interest j given all remaining features $-j$. Thus, it is more difficult to estimate than its marginal-sampling-based counterpart, Permutation Feature Importance (PFI); Marginal sampling can be performed by randomly drawing values from the feature of interest's observation vector.

We propose a novel algorithm to estimate CFI. It is based on the assumption that we can learn a partitioning of the feature space, such that in each subgroup the feature of interest j is independent of the remaining variables. As a consequence of the independence, in each subgroup, marginal and conditional sampling coincide. Thus, the partitioning allows us to sample from the conditional distribution by permuting observations within each group. To learn the partitioning, we leverage tree-based algorithms such as CART and transformation trees.

In settings where the learned partitioning is sparse, the subgroup-based approach has a further advantage (beyond good approximation to the true CFI values). Shallow trees are easy to interpret, such that the partitioning itself is interpretable. As such, we can generate subgroup-specific interpretations. For instance, by applying PFI in each subgroup, we find that temperature is predictive of bike rentals in summer, but not in winter.

Similarly, we propose a subgroup-based variant of PDPs. PDPs rely on marginal sampling and therefore may evaluate the model on unrealistic observations and thus provides limited insight into the data. If applied within the subgroups, where the feature of interest is independent of the remaining features, no dependencies between features are destroyed, and inference about the DGP is possible. For instance, by inspecting the PDP in the subgroups we may learn how the conditional expectation of the target varies with temperature given that we know that it's winter.

Chapter 2

Contributed Articles

2.1 Paper I: Improvement-focused Causal Recourse (ICR)

König, Gunnar, Timo Freiesleben, and Moritz Grosse-Wentrup. **Improvement-Focused Causal Recourse (ICR).** *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 37. No. 10. 2023.

Gunnar König contributed to the paper as first author. Gunnar König had the initial idea, wrote large parts of the paper, developed the proofs and wrote the code. Timo Freiesleben helped to develop the story and the philosophical foundation, wrote large parts of Section 4, checked the proofs and contributed to Sections 1, 2, 9 and 10. All authors helped to revise and proofread the paper.

For legal reasons the arXiv preprint of the article is included in this document (https://doi.org/10.48550/arXiv.2210.15709). The original version can be accessed via https://doi.org/10.1609/aaai.v37i10.26398.

IMPROVEMENT-FOCUSED CAUSAL RECOURSE (ICR)

Gunnar König[1,2,3], **Timo Freiesleben**[4,5], **and Moritz Grosse-Wentrup**[2]

[1]Institute for Statistics, LMU Munich
[2]Research Group Neuroinformatics, University of Vienna
[3]Munich Center for Machine Leanring (MCML)
[4]Cluster of Excellence: Machine Learning for Science, University of Tübingen
[5]Munich Center for Mathematical Philosophy (MCMP), LMU Munich

ABSTRACT

Algorithmic recourse recommendations, such as Karimi et al.'s (2021) causal recourse (CR), inform stakeholders of how to act to revert unfavorable decisions. However, there are actions that lead to acceptance (i.e., revert the model's decision) but do not lead to improvement (i.e., may not revert the underlying real-world state). To recommend such actions is to recommend fooling the predictor. We introduce a novel method, Improvement-Focused Causal Recourse (ICR), which involves a conceptual shift: Firstly, we require ICR recommendations to guide towards improvement. Secondly, we do not tailor the recommendations to be accepted by a specific predictor. Instead, we leverage causal knowledge to design decision systems that predict accurately pre- and post-recourse. As a result, improvement guarantees translate into acceptance guarantees. We demonstrate that given correct causal knowledge ICRguides towards both acceptance and improvement.

Keywords algorithmic recourse · gaming · causal inference · interpretable machine learning · robustness

1 Introduction

Predictive systems are increasingly deployed for high-stakes decisions, for instance in hiring [Raghavan et al., 2020], judicial systems [Zeng et al., 2017], or when distributing medical resources [Obermeyer and Mullainathan, 2019]. A range of work [Wachter et al., 2017, Ustun et al., 2019, Karimi et al., 2021] develops tools that offer individuals possibilities for so-called algorithmic recourse (i.e. actions that revert unfavorable decisions). Joining previous work in the field, we distinguish between reverting the model's prediction $\hat{Y}$ (acceptance) and reverting the underlying real-world state Y (improvement) and argue that recourse should lead to acceptance *and improvement* [Ustun et al., 2019, Barocas et al., 2020]. Existing methods, such as counterfactual explanations (CE; Wachter et al. [2017]) or causal recourse (CR; Karimi et al. [2021]), ignore the underlying real-world state and only optimize for acceptance. Since ML models are not designed to predict accurately in interventional environments (i.e. environments where actions have changed the data distribution), acceptance does not necessarily imply improvement.

Let us consider a simple motivational example. The goal is to predict whether hospital visitors without recent test certificate are infected with Covid in order to restrict access to tested and low-risk individuals. In the example, the model's *prediction* $\hat{Y}$ represents whether someone is classified to be infected, whereas the *prediction target* Y represents whether someone is actually infected. Target and prediction differ in how they are affected by actions. E.g., intervening on the *symptoms* may change the diagnosis $\hat{Y}$, but will not affect whether someone is infected (Y).

Both counterfactual explanations (CE) and causal recourse (CR) only target $\hat{Y}$ (Figure 1). Therefore, CE and CR may suggest to alter the *symptoms* (e.g., by taking cough drops) and thereby may recommend to *game* the predictor: Although the intervention leads to acceptance the actual Covid risk Y is not improved.[1]

One may argue that this is an issue of the prediction model and may adapt the predictor strategically to make gaming less lucrative than improvement [Miller et al., 2020]. In our example, the model's reliance on the symptom state would need to be reduced. However, such strategic adaptions may come at the cost of predictive performance since gameable variables, like the symptom state, can be highly predictive [Shavit et al., 2020]. Thus, we tackle the problem by adjusting the explanation.

[1] In E.1, the case is formally demonstrated.

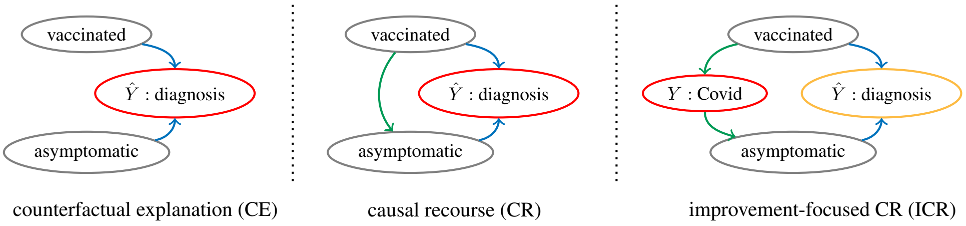


Figure 1: Directed Acyclic Graph (DAG) illustrating the perspective on model and data taken by counterfactual explanations (CE, left) and causal recourse (CR, center) in contrast to improvement-focused recourse (ICR, right). Blue edges represent the causal links induced by the prediction model, green edges the real-world causal links, gray nodes the covariates, and the red (yellow) node the primary (secondary) recourse target. CR respects the causal relationships but only between input features. ICR is the only approach that takes the target Y into account. While CE and CR aim to revert the prediction $\hat{Y}$, ICR aims to revert the target Y.

Contributions We present improvement-focused causal recourse (ICR), the first recourse method that targets improvement instead of acceptance. Since estimating the effects of actions is a causal problem, causal knowledge is required. More specifically, we show how to exploit either knowledge of the structural causal model (SCMs) or the causal graph to guide towards improvement (Section 5). On a conceptual level we argue that the individual's improvement options should not be limited by an acceptance constraint (Section 4). In order to nevertheless yield acceptance, we show how to exploit said causal knowledge to design post-recourse decision systems that in expectation recognize improvement (Section 6), such that improvement guarantees translate into acceptance guarantees (Section 7). On synthetic and semi-synthetic data, we demonstrate that ICR, in contrast to existing approaches, leads to improvement and acceptance (Section 8).

2 Related Work

Constrastive Explanations Contrastive explanations explain decisions by contrasting them with alternative decision scenarios [Karimi et al., 2020a, Stepin et al., 2021]; a well known example are counterfactual explanations (CE) that highlight the minimal feature changes required to revert the decision of a predictor $\hat{f}(x)$ [Wachter et al., 2017, Dandl et al., 2020]. However, CEs are ignorant of causal dependencies in the data and therefore in general fail to guide action [Karimi et al., 2021]. In contrast, the causal recourse (CR) framework by Karimi et al. [2022] takes the causal dependencies between covariates into account: More specifically, Karimi et al. [2022] use structural causal models or causal graphs to guide individuals towards acceptance.[2] The importance of improvement was discussed before [Ustun et al., 2019, Barocas et al., 2020], but as of now no improvement-focused recourse method was proposed.

Strategic Classification The related field of strategic modeling investigates how the prediction mechanism incentivizes rational agents [Hardt et al., 2016, Tsirtsis and Gomez Rodriguez, 2020]. A range of work [Bechavod et al., 2020, Chen et al., 2020, Miller et al., 2020] thereby distinguishes models that incentivize *gaming* (i.e., interventions that affect the prediction $\hat{Y}$ but not the underlying target Y in the desired way) and *improvement* (i.e., actions that also yield the desired change in Y). Strategic modeling is concerned with adapting the model, where except for special cases the following three goals are in conflict: incentivizing improvement, predictive accuracy, and retrieving the true underlying mechanism [Shavit et al., 2020].

Robust algorithmic recourse The robustness of CEs and CR has been investigated before [Rawal et al., 2021, Pawelczyk et al., 2020, Upadhyay et al., 2021, Dominguez-Olmedo et al., 2021, Pawelczyk et al., 2022], yet only with respect to generic shifts of model and data. Only Pawelczyk et al. [2020] investigate the robustness regarding refits on the same data. They find that on-the-manifold CEs are more robust than standard CEs. In contrast, we empirically compare the robustness of CE, CR and ICR with respect to refits on the same data.

3 Background and Notation

Prediction model We assume binary probabilistic predictors and cross-entropy loss, such that the optimal score function $h^*(x)$ models the conditional probability $P(Y = 1|X = x)$, which we abbreviate as $p(y|x)$. We denote the

[2] For the interested reader, we formally introduce CR in our notation in A.4.

estimated score function as $\hat{h}(x)$, which can be transformed into the binary decision function $\hat{f}(x) := [\hat{h}(x) \geq t]$ via the decision threshold t.

Causal data model We model the data generating process using a structural causal model (SCM) $\mathcal{M} \in \Pi$ [Pearl, 2009, Peters et al., 2017]. The model $\mathcal{M} = \langle X, U, \mathbb{F} \rangle$ consists of the endogenous variables $X \in \mathcal{X}$, the mutually independent exogenous variables $U \in \mathcal{U}$, and structural equations $\mathbb{F} : \mathcal{U} \rightarrow \mathcal{X}$. Each structural equation f_j specifies how X_j is determined by its endogenous causes and the corresponding exogenous variable U_j. The SCM entails a directed graph $\mathcal{G}$, where variables are connected to their direct effects via a directed edge.
The index set of endogenous variables is denoted as D. The parent indexes of node j are referred to as $pa(j)$ and the children indexes as $ch(j)$. We refer to the respective variables as $X_{pa(j)}$. We write $X_{pa(j)}$ to denote all parents excluding Y and $(X, Y)_{pa(j)}$ to denote all parents including Y. All ascendant indexes of a set S are denoted as $asc(S)$, its complement as $nasc(S)$, all descendant indexes as $d(S)$, and its complement as $nd(S)$.
SCMs allow to answer causal questions. This means that they cannot only be used to describe (conditional) distributions (observation, rung 1 on Pearl's ladder of causation [Pearl, 2009]), but can also be used to predict the (average) effect of actions $do(x)$ (intervention, rung 2) and imagine the results of alternative actions in light of factual observation $(x, y)^F$ (counterfactuals, rung 3).
As such, we model actions as structural interventions $a : \Pi \rightarrow \Pi$, which can be constructed as $do(a) = do(\{X_i := \theta_i\}_{i \in I})$, where I is the index set of features to be intervened upon. A model of the interventional distribution can be obtained by fixing the intervened upon values to θ_I (e.g. by replacing the structural equation $f_I := \theta_I$). Counterfactuals can be computed in three steps [Pearl, 2009]: First, the factual distribution of exogenous variables U given the factual observation of the endogenous variables x^F is inferred (*abduction*) (i.e., $P(U_j|X^F)$). Second, the structural interventions corresponding to $do(a)$ are performed (*action*). Finally, we can sample from the counterfactual distribution $P(X^{SCF}|X = x^F, do(a))$ using the abducted noise and the intervened-upon structural equations (*prediction*).

4 The Two Tales of Contrastive Explanations

In the introduction we have demonstrated that CE and CR may suggest to game the predictor (i.e. guide towards acceptance without improvement). To tackle the issue, we will introduce a new explanation technique called improvement-focused causal recourse (ICR) in Section 5.
In this section we lay the conceptual justification for our method. More specifically, we argue that for recourse the acceptance constraint of CR should be *replaced* by an improvement constraint. Therefore, we first recall that a multitude of goals may be pursued with contrastive explanations [Wachter et al., 2017] and separate two purposes of contrastive explanations: *contestability of algorithmic decisions* and *actionable recourse*. We then argue that improvement is an essential requirement for recourse and that the individual's options for improvement should not be limited by acceptance constraints.

Contestability and recourse are distinct goals. *Contestability* is concerned with the question of whether the algorithmic decision is correct according to common sense, moral or legal standards. Explanations may help model authorities to detect violations of such standards or enable explainees to contest unfavorable decisions [Wachter et al., 2017, Freiesleben, 2021]. Explanations that aim to enable contestability must reflect the model's rationale for an algorithmic decision. *Recourse recommendations* on the other hand need to satisfy various constraints unrelated to the model, such as causal links between variables [Karimi et al., 2021] or their actionability [Ustun et al., 2019]. Consequently, explanations geared to contest are more complete and true to the model while recourse recommendations are more selective and true to the underlying process.[3] We believe that the selectivity and reliance of recourse recommendations on factors besides the model itself is not a limitation but an indispensable condition for making explanations more relevant to the explainee.

In the context of recourse, improvement is desirable for model authority and explainee. We consider improvement to be an important normative requirement for recourse, both with respect to explainee and model authority. Valuable recourse recommendations enable explainees to plan and act; thus, such recommendations must either provide indefinite validity or a clear expiration date [Wachter et al., 2017, Barocas et al., 2020, Venkatasubramanian and Alfano, 2020]. Problematically, when model authorities give guarantees for non-improving recourse, this constitutes a binding commitment to misclassification. However, if model authorities do not provide recourse guarantees over time, this diminishes the value of recourse recommendations to explainees. They might invest effort into non-improving actions

[3] We do not claim that recourse and contestability always diverge, we only describe a difference in focus. If contesting is successful it may even provide an alternative route towards recourse.

that ultimately do not even lead to acceptance because the classifier changed.[4] In contrast, improvement-focused recourse is honored by any accurate classifier. We conclude that, given these advantages for both model authority and explainee, recourse recommendations should help to improve the underlying target Y.[5]

Improvement should come first, acceptance second. Taken that we constrain the optimization on improvement, how to guarantee acceptance remains an open question. One approach would be to constrain the optimization on both improvement and acceptance. However, a restriction on acceptance is either redundant or, from our moral standpoint, questionable: If improvement already implies acceptance, the constraint is redundant. In the remaining cases, we can predict improvement with the available causal knowledge but would withhold these (potentially less costly) improvement options because of the limitations of the observational predictor. To ensure that acceptance ensues improvement, we instead suggest to exploit the assumed causal knowledge for accurate post-recourse prediction (Section 6), such that acceptance guarantees can be made (Section 7).

5 Improvement-Focused Causal Recourse (ICR)

We continue with the formal introduction of ICR, an explanation technique that targets improvement ($Y = 1$) instead of acceptance ($\hat{Y} = 1$). Therefore we first define the improvement confidence γ, which can be optimized to yield ICR. Like previous work in the field [Karimi et al., 2020b], we distinguish two settings: In the first setting, knowledge of the SCM can be assumed, such that we can leverage structural counterfactuals (rung 3 on Pearl's ladder of causation) to introduce the individualized improvement confidence γ^{ind}. In the second setting only the causal graph is known, which we exploit to propose the subpopulation-based improvement confidence γ^{sub} (rung 2).

Individualized improvement confidence For the individualized improvement confidence γ^{ind} we exploit knowledge of a SCM. SCMs can be used to answer counterfactual questions (rung 3). In contrast to rung-2-predictions, counterfactuals are tailored to the individual and their situation [Pearl, 2009]: They ask what would have been if one had acted differently and thereby exploit the individual's factual observation. Given unchanged circumstances, counterfactuals can be seen as individualized causal effect predictions.

In contrast to existing SCM-based recourse techniques [Karimi et al., 2022] we include both the prediction $\hat{Y}$ and the target variable Y as separate variables in the SCM. As a result, the SCM can be used not only to model the individualized probability of acceptance, but also the individualized probability of improvement.

Definition 1 (Individualized improvement confidence)**.** *For pre-recourse observation x^{pre} and action a we define the individualized improvement confidence as*

$$\gamma^{ind}(a) = \gamma(a, x^{pre}) := P(Y^{post} = 1 | do(a), x^{pre}).$$

Since the pre-recourse (factual) target Y cannot be observed, standard counterfactual prediction cannot be applied directly. However, we can regard the distribution as a mixture with two components, one for each possible state of Y. We can estimate the mixing weights using h^* and each component using standard counterfactual prediction. Details including pseudocode are provided in B.1.

Subpopulation-based improvement confidence For the estimation of the individualized improvement confidence γ^{ind} knowledge of the SCM is required. If the SCM is not specified, but the causal graph is known instead and there are no unobserved confounders (causal sufficiency), we can still estimate the effect of interventions (rung 2).

In contrast to counterfactual distributions (rung 3), interventional distributions describe the whole population and therefore provide limited insight into the effects of actions on specific individuals. Building on Karimi et al. [2020b], we thus narrow the population down to a subpopulation of similar individuals, for which we then estimate the subpopulation-based causal effect. More specifically, we consider individuals to belong to the same subgroup if the variables that are not affected by the intervention take the same values. For action a, we define the subgroup characteristics as $G_a := nd(I_a)$ (i.e., the non-descendants of the intervened-upon variables in the causal graph).[6] More formally, we define the subpopulation-based improvement confidence γ^{sub} as the probability of Y taking the favorable outcome in the subgroup of similar individuals (Definition 2).

[4] For instance, in the introductory example, an intervention on the symptom state would only be honored by a refit of the model on pre- and post-recourse data for the small percentage of individuals who were already vaccinated, as documented in more detail in E.1. Also, gaming actions may not be robust concerning model multiplicity, as seen in the experiments (Section 8).

[5] We do not claim that gaming is necessarily bad; it may be justified when predictors perform morally questionable tasks.

[6] The estimand resembles the conditional treatment effect with G_a being effect modifiers [Hernán MA, 2020].

Definition 2 (Subpopulation-based improvement confidence)**.** *Let a be an action that potentially affects Y, i.e. $I_a \cap asc(Y) \neq \emptyset$.*[7] *Then we define the subpopulation-based improvement confidence as*

$$\gamma^{sub}(a) = \gamma(a, x^{pre}_{G_a}) := P(Y^{post} = 1 | do(a), x^{pre}_{G_a}).$$

The set G_a is chosen for practical reasons. In order to make the estimation more accurate, we would like to condition on as many characteristics as possible. However, without access to the SCM, one can only identify interventional distributions for subgroups of the population by conditioning on their (unobserved) post-intervention characteristics (but not by conditioning on their pre-intervention characteristics) [Pearl, 2009, Glymour et al., 2016]. If we were to select a subgroup from a post-recourse distribution by conditioning on pre-recourse characteristics that are affected by a (e.g. strong pre-recourse symptoms), we yield a group that the individual may not be part of (e.g. people with strong post-recourse symptoms). In contrast, for X_{G_a} pre- and post-intervention values coincide, such that we can estimate γ^{sub}: Assuming causal sufficiency, the standard procedure to sample interventional distributions can be applied, only that additionally $X^{post}_{G_a} := x^{pre}_{G_a}$. Based on the sample γ^{sub} can be estimated (as detailed in B.3).
The estimation of γ^{sub} does not require knowledge of the SCM, but is less accurate than γ^{ind}. In the introductory example, for the action *get vaccinated* the set of subgroup-characteristics G_a is empty. As such, γ^{sub} is concerned with the effect of a vaccination over the whole population. If we were to observe *zip code*, a variable that is not affected by *vaccination*, γ^{sub} would indicate the effect of vaccination for subjects that share the explainee's *zip code*. In contrast, γ^{ind} also takes the explainee's *symptom state* into account.

Optimization problem To generate ICR recommendations, we can optimize Equation 1. We aim to find actions that meet a user-specified improvement target confidence $\overline{\gamma}$ with minimal cost for the recourse seeking individual. The cost function cost(a, x^{pre}) captures the effort the individual requires to perform action a [Karimi et al., 2020b].
As for CE or CR, the optimization problem for ICR is computationally challenging (B.4). It can be seen as a two-level problem, where on the first level the intervention targets I_a, and on the second level the corresponding intervention values θ_a are optimized [Karimi et al., 2020b]. Since we target improvement, we can restrict I_a to causes of Y. Following Dandl et al. [2020], we use the genetic algorithm NSGA-II [Deb et al., 2002] for optimization.

$$\text{argmin}_{a=do(X_I=\theta)} \quad \text{cost}(a, x^{pre}) \quad \text{s.t.} \quad \gamma(a) \geq \overline{\gamma}. \tag{1}$$

6 Accurate Post-Recourse Prediction

Recourse recommendations should not only lead to improvement Y but also revert the decision $\hat{Y}$. Whether acceptance guarantees naturally ensue from γ depends on the ability of the predictor to recognize improvements. As follows, we demonstrate how the assumed causal knowledge can be exploited to design accurate post-recourse predictors. We find that an individualized post-recourse predictor is required to translate γ^{ind} into an individualized acceptance guarantee, but curiously that the observational predictor is sufficient in supopulation-based settings.

Individualized post-recourse prediction If we were to use the optimal pre-recourse observational predictor h^* for post-recourse prediction, there would be an imbalance in predictive capability between ML model and individualized ICR: ICR individualizes its predictions using x^{pre} and the SCM. This knowledge is not accessible by the predictor h^*, which only makes use of x^{post}. As such, improvement that was accurately predicted by ICR is not necessarily recognized by h^* and γ^{ind} cannot be directly translated into an acceptance bound. We demonstrate the issue at an Example in E.3.[8]
In order to settle the imbalance between ICR and the predictor, we suggest to leverage the SCM not only when generating individualized ICR recommendations but also when predicting post-recourse, such that the predictor is at least as accurate as γ^{ind}. More formally, we suggest to estimate the post-recourse distribution of Y conditional on x^{pre}, $do(a)$, and the post-recourse observation $x^{post,a}$ (Definition 3). This post-recourse prediction resembles the counterfactual distribution, except that we additionally take the factual post-recourse observation of the covariates into account.

[7] If a cannot affect Y, we can predict $P(Y|x^{pre}, do(a)) = P(Y|x^{pre})$ using the optimal observational predictor h^*.

[8] One may also argue that standard predictive models are not suitable since optimality of the predictor in the pre-recourse distribution does not necessarily imply optimality in interventional environments (as Example 1, E.1 demonstrates). We can refute this criticism using Proposition 3, where we learn that $\hat{h}^*$ is stable with respect to ICR actions.

Definition 3 (Individualized post-recourse predictor)**.** *We define the individualized post-recourse predictor as*

$$h^{*,ind}(x^{post}) = P(Y^{post} = 1|x^{post}, x^{pre}, do(a))$$

For SCMs with invertible equations, $h^{*,ind}$ can be estimated using a closed form solution. Otherwise we can sample from the counterfactual post-recourse distribution $p(y^{post}, x^{post}|x^{pre}, do(a))$ (as we did for the estimation of γ^{ind}), select the samples that conform with x^{post} and compute the proportion of favorable outcomes (details in B.2).

For the individualized post-recourse predictor, improvement probability and prediction are closely linked (Proposition 1). More specifically, the expected post-recourse prediction $h^{*,ind}$ is equal to the individualized improvement probability $\gamma(x^{pre}, a)$. We will exploit Proposition 1 in Section 7, where we derive acceptance guarantees for ICR.

Proposition 1. *The expected individualized post-recourse score is equal to the individualized improvement probability $\gamma^{ind}(x^{pre}, a) := P(Y^{post} = 1|x^{pre}, do(a))$, i.e.*

$$E[\hat{h}^{*,ind}(x^{post})|x^{pre}, do(a)] = \gamma^{ind}(a).$$

Subpopulation-based post-recourse prediction Curiously we find that for ICR actions a the optimal observational pre-recourse predictor h^* remains accurate: in the subpopulation of similar individuals the expected post-recourse prediction corresponds to the improvement probability $\gamma^{sub}(a)$ (Proposition 3). This allows us to derive acceptance guarantees for h^* in Section 7.

This result is in contrast to the negative results for CR, where actions may not affect prediction and the underlying target coherently, such that the predictive performance deteriorates (as demonstrated in the introduction, and more formally in E.1). The key difference to CR is that ICR actions exclusively intervene on causes of Y: Interventions on non-causal variables may lead to a shift in the conditional distribution $P(Y|X_S)$ (where $S \subseteq D$ is any set of variables that allows for optimal prediction). In contrast, given causal sufficiency, the conditional $P(Y|X_S)$ is stable to interventions on causes of Y.

Proposition 2. *Given nonzero cost for all interventions, ICR exclusively suggests actions on causes of Y. Assuming causal sufficiency, for optimal models the conditional distribution of Y given the variables X_S that the model uses (i.e. $P(Y|X_S)$) is stable w.r.t interventions on causes. Therefore, optimal predictors are intervention stable w.r.t. ICR actions.*

Proposition 3. *Given causal sufficiency and positivity*[9]*, for interventions on causes the expected subgroup-wide optimal score h^* is equal to the subgroup-wide improvement probability $\gamma^{sub}(a) := P(Y^{post} = 1|do(a), x^{pre}_{G_a})$, i.e.*

$$E[\hat{h}^*(x^{post})|x^{pre}_{G_a}, do(a)] = \gamma^{sub}(a).$$

Link between CR and ICR: Proposition 2 has further interesting consequences. For CR actions a that only intervene on causes of Y and that are guaranteed to yield a predicted score ζ in the subpopulation, we can infer that $\gamma^{sub}(a) \geq \zeta$. For instance, if acceptance with respect to a 0.5 decision threshold can be guaranteed, that implies improvement with at least 50% probability. As such, in subpopulation-based settings (1) improvement guarantees can be made for CR if only interventions on causes are lucrative, and (2) CR can be adapted to also guide towards improvement by a restricting actions to intervene on causes.

7 Acceptance Guarantees

For the presented accurate post-recourse predictors, improvement guarantees translate into acceptance guarantees (Proposition 4). The reason is that the post-recourse prediction is linked to γ (Propositions 1 and 3).

Proposition 4. *Let g be a predictor with $E[g(x^{post})|x^{pre}_S, do(a)] = \gamma(x^{pre}_S, a)$. Then for a decision threshold t the post-recourse acceptance probability $\eta(t; x^{pre}_S, a) := P(g(x^{post}) > t|x^{pre}_S, do(a))$ is lower bounded by the respective improvement probability:*

$$\eta(t; x^{pre}_S, a, g) \geq \frac{\gamma(x^{pre}_S, a) - t}{1 - t}.$$

Proof (sketch): We decompose the expected prediction (γ) into true positive rate (TPR), false negative rate (FNR) and acceptance rate. By bounding TPR and FNR we yield the presented acceptance bound. The proof is provided in D.4.

[9] Positivity ensures that the post-recourse observation lies within the observational support [Neal, 2020], where the model was trained (i.e., $p^{pre}(x^{post}) > 0$)).

Using Proposition 4, we can tune confidence γ and the model's decision threshold to yield a desired acceptance rate. For instance, we can guarantee acceptance with (subgroup-wide) probability $\eta \geq 0.9$ given $\gamma = 0.95$ and a global decision threshold $t = 0.5$.
Furthermore we can leverage the sampling procedures that we use to compute γ to estimate the individualized or subpopulation-based acceptance rate $\eta(t; x_S^{pre}, a, g)$ (as detailed in B.1 and B.3). To guarantee acceptance with certainty, the decision threshold can be set to $t = 0$.
For the explainee, it is vital that the acceptance guarantee is presented in a human-intelligible fashion. In contrast to previous work in the field, we suggest to communicate the acceptance guarantee in terms of a probability.[10] Furthermore, for subpopulation-based recourse, the set of subgroup characteristics should be transparent. In the hospital admission example, the subpopulation-based acceptance guarantee could be communicated as follows: *Within a group of individuals that share your zip code, a vaccination leads to acceptance with at least probability η.*

8 Experiments

In the experiments we evaluate the following questions, assuming correct causal knowledge and accurate models of the conditional distributions in the data:

Q1: Do CE, CR and ICR lead to improvement?
Q2: Do CE, CR and ICR lead to acceptance (by pre- and post- post-recourse predictor)?
Q3: Do CE, CR and ICR lead to acceptance by other predictors with comparable test error?[11]
Q4: How costly are CE, CR and ICR recommendations?

Setup We evaluate CE, individualized and subpopulation-based CR and ICR with various confidence levels, over multiple runs, and on multiple synthetic and semi-synthetic datasets with known ground-truth (listed below).[12] Random forests were used for prediction, except in the *3var* settings where logistic regression models were used. Following Dandl et al. [2020], we use NSGA-II [Deb et al., 2002] for optimization. For a full specification of the SCMs including the linear cost functions we refer to C.2. Details on the implementation and access to the code are provided in C.1.

3var-causal: A linear gaussian SCM with binary target Y, where all features are causes of Y.
3var-noncausal: The same setup as *3var-causal*, except that one of the features is an effect of Y.
5var-skill: A categorical semi-synthetic SCM where programming skill-level is predicted from causes (e.g. *university degree*) and non-causal indicators extracted from GitHub (e.g. *commit count*).
7var-covid: A semi-synthetic dataset inspired by a real-world covid screening model [Jehi et al., 2020, Wynants et al., 2020].[13] The model includes typical causes like *covid vaccination* or *population density* and symptoms like *fever* and *fatigue*. The variables are mixed categorical and continuous with various noise distributions. Their relationships include nonlinear structural equations.

Results The results are visualized in Figure 2 and provided in tabular form in C.3.

Q1 (Figure 2a): In scenarios where gaming is possible and lucrative (*3var-noncausal*, *5var-skill* and *7var-covid*) ICR reliably guides towards improvement, but CE and CR game the predictor and yield improvement rates close to zero. For instance, on *5var-skill* CE and CR exclusively suggest to tune the GitHub profile (e.g. by adding more commits). Since the employer offered recourse it should be honored although the applicants remain unqualified. In contrast, ICR suggests to get a degree or to gain experience, such that recourse implementing individuals are suited for the job.
On *3var-causal*, where gaming is not possible, CR also achieves improvement. However, since acceptance w.r.t to a decision treshold $t = 0.5$ is targeted, only improvement rates close to 50% are achieved (the expected predicted score translates into γ^{sub} (Proposition 3)).
For subp. ICR, γ^{obs} is below $\overline{\gamma}$, because the subpopulation may include individuals that were already accepted pre-recourse, such that γ^{sub} and γ^{obs} may not coincide.

Q2 (Figure 2d): All methods yield the desired acceptance rates w.r.t. to the pre-recourse predictor.[14] For CE and CR η^{obs} is higher than for ICR, and for ind. recourse higher than for subp. recourse. Curiously, although no acceptance

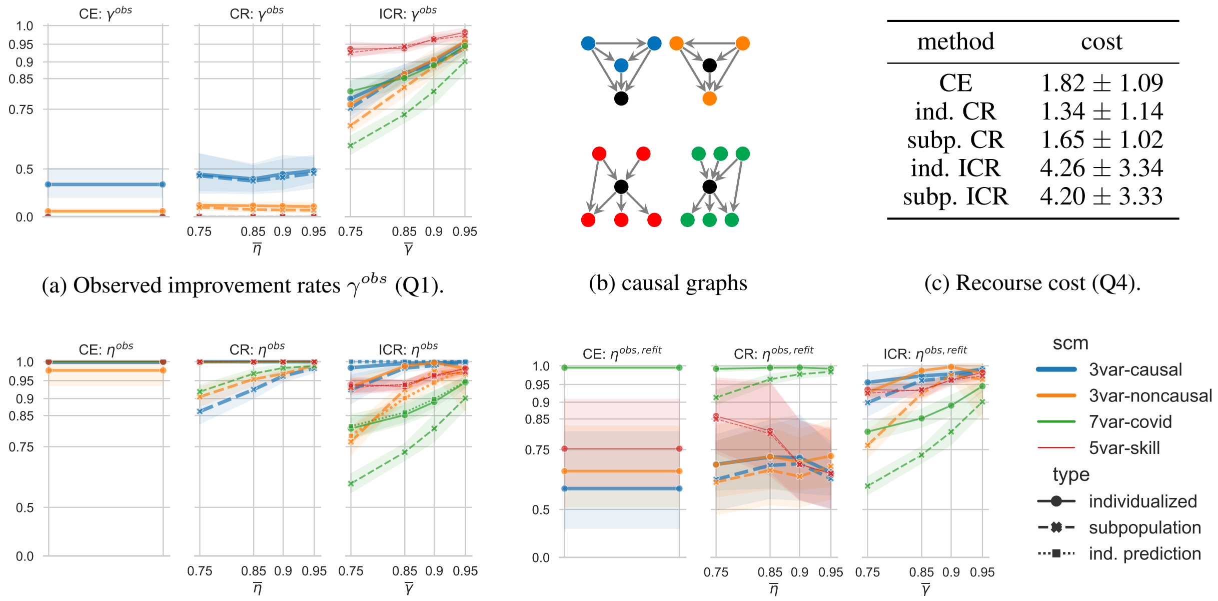
[10] For CR, the acceptance confidence is encoded in a hyperparameter, as explained in E.2.

[11] The problem that refits on the same data with similar performance have different mechanism is known as the Rashomon problem or model multiplicity [Breiman, 2001, Pawelczyk et al., 2020, Marx et al., 2020].

[12] For ground-truth counterfactuals, simulations are necessary [Holland, 1986].

[13] The real-world screening model is used to decide whether individuals need a test certificate to enter a hospital. It can be accessed via `https://riskcalc.org/COVID19/`.

[14] ICR holds the acceptance rates from Proposition 4, as analyzed in more detail in C.3.



(a) Observed improvement rates γ^{obs} (Q1).

(b) causal graphs

method	cost
CE	1.82 ± 1.09
ind. CR	1.34 ± 1.14
subp. CR	1.65 ± 1.02
ind. ICR	4.26 ± 3.34
subp. ICR	4.20 ± 3.33

(c) Recourse cost (Q4).

(d) Observed acceptance rates η^{obs} w.r.t. h^*; for ind. ICR additionally w.r.t. $h^{*,ind}$ (Q2).

(e) Observed acceptance rates for other fits with comparable test set performance $\eta^{obs,\text{refit}}$ (Q3).

(f) legend

Figure 2: Experimental results for CE, CR and ICR on four datasets over 10 runs on 200 individuals each. For the probabilistic methods the confidences $0.75, 0.85, 0.9, 0.95$ were targeted (for CR: $\overline{\eta}$, for ICR: $\overline{\gamma}$). For CE no slack is allowed, such that the results correspond to a confidence level of 1.0. Values are reported on a quadratic scale.

guarantees could be derived for the pre-recourse predictor and ind. ICR, we find that both pre- and ind. post-recourse predictor reliably lead to acceptance.[15]

Q3 (Figure 2e): We observe that CE and CR actions are unlikely to be honored by other model fits with similar performance on the same data. This result is highly relevant to practitioners, since models deployed in real-world scenarios are regularly refitted. As such, individuals that implemented acceptance-focused recourse may not be accepted after all, since the decision model was refitted in the meantime. In contrast, ICR acceptance rates are nearly unaffected by refits. The result confirms our argument that improvement-focused recourse may be more desirable for explainees (Section 4).

Q4 (Table 2c): CR actions are cheaper than ICR actions, since improvement may require more effort than gaming. As such, CR has benefits for the explainee: For instance, on *5var-skill*, CR suggests to tune the GitHub profile (e.g. by adding more commits), which requires less effort than earning a degree or gaining job experience. Detailed results on cost are reported in C.3.

In conclusion, ICR actions require more effort than CR, but lead to improvement and acceptance while being more robust to refits of the model.

9 Limitations and Discussion

Causal knowledge and assumptions Individualized ICR requires a fully specified SCM; Subpopulation-based ICR is less demanding but still requires the causal graph and causal sufficiency. SCMs and causal graphs are rarely readily available in practice [Peters et al., 2017] and causal sufficiency is difficult to test [Janzing et al., 2012]. Research on causal inference gives reason for cautious optimism that the difficulties in constructing SCMs and causal graphs can eventually be overcome [Spirtes and Zhang, 2016, Peters et al., 2017, Heinze-Deml et al., 2018, Malinsky and Danks, 2018, Glymour et al., 2019].
There are further foundational problems linked to causality that affect our approach: causal cycles, an ontologically vague target Y (e.g. in hiring), disparities in our data, or causal model misspecification [Barocas and Selbst, 2016,

[15] Given that the ind. post-recourse predictor is much more difficult to estimate, the pre-recourse predictor in combination with individualized acceptance guarantees (B.1) may cautiously be used as fallback.

Barocas et al., 2017, Bongers et al., 2021]. All of these factors are considered difficult open problems and may have detrimental impact on our, as well as on any other, recourse framework.
Guiding action without causal knowledge is impossible; when causal knowledge is available, our work provides a normative framework for improvement-focused recourse recommendations. Thus, we join a range of work in explainability [Frye et al., 2020, Heskes et al., 2020, Wang et al., 2021, Zhao and Hastie, 2021] and fairness [Kilbertus et al., 2017, Kusner et al., 2017, Zhang and Bareinboim, 2018, Makhlouf et al., 2020] that highlights the importance of causal knowledge.

Contestability Improvement-focused recourse guides individuals towards actions that help them to improve, e.g., it recommends a vaccination to lower the risk to get infected with Covid. If, however, a explainee is more interested in contesting the algorithmic decision, (improvement-focused) recourse recommendations are not sufficient. Think of an individual who is denied entrance to an event because of their high Covid risk prediction, which is based on a non-causal, spurious association with their country of origin[16]. In such situations, we suggest to additionally show explainees diverse explanations, which enable to contest the decision. For example, such an explanation could be: if your country of origin would be different, your predicted Covid risk would have been lower.

10 Conclusion

In the present paper, we took a causal perspective and investigated the effect of recourse recommendations on the underlying target variable. We demonstrated that acceptance-focused recourse recommendations like counterfactual explanations or causal recourse may not improve the underlying prediction but game the predictor instead. The problem stems from predictive, but non-causal relationships, which are abundant in machine learning applications.[17]
We tackled the problem in the explanation domain and introduced Improvement-Focused Causal Recourse (ICR), an explanation technique that guides towards improvement of the prediction target and demonstrated how to design post-recourse predictors such that improvement leads to acceptance. We confirm the theoretical results in experiments. With ICR we hope to inspire a shift from acceptance- to improvement-focused recourse.

Acknowledgements

This work was supported by the Graduate School of Systemic Neuroscience (GSN) of the LMU Munich and by the German Federal Ministry of Education and Research (BMBF).

[16] E.g., due to a spurious association with the causal variable *type of vaccine*.

[17] For instance, in hiring, certain keywords in the CV may be associated with qualification, but adding them to the CV does not improve aptitude [Strong, 2022].

References

Manish Raghavan, Solon Barocas, Jon Kleinberg, and Karen Levy. Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 469–481, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369367.

Jiaming Zeng, Berk Ustun, and Cynthia Rudin. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 180(3):689–722, 2017.

Ziad Obermeyer and Sendhil Mullainathan. Dissecting racial bias in an algorithm that guides health decisions for 70 million people. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 89–89, 2019.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

Berk Ustun, Alexander Spangher, and Yang Liu. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 10–19, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361255.

Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse: From counterfactual explanations to interventions. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 353–362, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383097.

Solon Barocas, Andrew D. Selbst, and Manish Raghavan. The hidden assumptions behind counterfactual explanations and principal reasons. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 80–89, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369367.

John Miller, Smitha Milli, and Moritz Hardt. Strategic classification is causal modeling in disguise. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6917–6926, Online, 13–18 Jul 2020. PMLR.

Yonadav Shavit, Benjamin Edelman, and Brian Axelrod. Causal strategic linear regression. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 8676–8686, virtual, 13–18 Jul 2020. PMLR.

Amir-Hossein Karimi, Gilles Barthe, Bernhard Schölkopf, and Isabel Valera. A survey of algorithmic recourse: definitions, formulations, solutions, and prospects. *arXiv preprint arXiv:2010.04050*, 2020a.

Ilia Stepin, Jose M Alonso, Alejandro Catala, and Martín Pereira-Fariña. A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access*, 9:11974–12001, 2021.

Susanne Dandl, Christoph Molnar, Martin Binder, and Bernd Bischl. Multi-objective counterfactual explanations. In Thomas Bäck, Mike Preuss, André Deutz, Hao Wang, Carola Doerr, Michael Emmerich, and Heike Trautmann, editors, *Parallel Problem Solving from Nature – PPSN XVI*, pages 448–469, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58112-1.

Amir-Hossein Karimi, Julius von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Towards causal algorithmic recourse. In *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*, pages 139–166. Springer, 2022.

Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM conference on innovations in theoretical computer science*, pages 111–122, 2016.

Stratis Tsirtsis and Manuel Gomez Rodriguez. Decisions, counterfactual explanations and strategic behavior. *Advances in Neural Information Processing Systems*, 33:16749–16760, 2020.

Yahav Bechavod, Katrina Ligett, Zhiwei Steven Wu, and Juba Ziani. Causal feature discovery through strategic modification. *arXiv preprint arXiv:2002.07024*, 2020.

Yatong Chen, Jialu Wang, and Yang Liu. Linear classifiers that encourage constructive adaptation. *arXiv preprint arXiv:2011.00355*, 2020.

Kaivalya Rawal, Ece Kamar, and Himabindu Lakkaraju. Algorithmic recourse in the wild: Understanding the impact of data and model shifts, 2021.

Martin Pawelczyk, Klaus Broelemann, and Gjergji. Kasneci. On counterfactual explanations under predictive multiplicity. In Jonas Peters and David Sontag, editors, *Proceedings of the 36th Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 124 of *Proceedings of Machine Learning Research*, pages 809–818, Online, 03–06 Aug 2020. PMLR.

Sohini Upadhyay, Shalmali Joshi, and Himabindu Lakkaraju. Towards robust and reliable algorithmic recourse. *Advances in Neural Information Processing Systems*, 34:16926–16937, 2021.

Ricardo Dominguez-Olmedo, Amir-Hossein Karimi, and Bernhard Schölkopf. On the adversarial robustness of causal algorithmic recourse. *arXiv preprint arXiv:2112.11313*, 2021.

Martin Pawelczyk, Teresa Datta, Johannes van-den Heuvel, Gjergji Kasneci, and Himabindu Lakkaraju. Algorithmic recourse in the face of noisy human responses. *arXiv preprint arXiv:2203.06768*, 2022.

Judea Pearl. *Causality*. Cambridge University Press, Cambridge, UK, 2 edition, 2009. ISBN 978-0-521-89560-6.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Timo Freiesleben. The intriguing relation between counterfactual explanations and adversarial examples. *Minds and Machines*, Oct 2021. ISSN 1572-8641.

Suresh Venkatasubramanian and Mark Alfano. The philosophical basis of algorithmic recourse. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 284–293, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369367.

Amir-Hossein Karimi, Julius von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse under imperfect causal knowledge: a probabilistic approach. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 265–277, virtual, 2020b. Curran Associates, Inc.

Robins JM Hernán MA. *Causal Inference: What If*. Boca Raton: Chapman & Hall/CRC, 2020.

Madelyn Glymour, Judea Pearl, and Nicholas P Jewell. *Causal inference in statistics: A primer*. John Wiley & Sons, 2016.

Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

Brady Neal. Introduction to causal inference from a machine learning perspective. *Course Lecture Notes (draft)*, 2020.

L Breiman. Statistical Modeling: The Two Cultures. *Statistical Science*, 16(3):199–231, 2001. ISSN 0889-5406.

Charles Marx, Flavio Calmon, and Berk Ustun. Predictive multiplicity in classification. In *International Conference on Machine Learning*, pages 6765–6774. PMLR, 2020.

Paul W Holland. Statistics and causal inference. *Journal of the American statistical Association*, 81(396):945–960, 1986.

Lara Jehi, Xinge Ji, Alex Milinovich, Serpil Erzurum, Brian P Rubin, Steve Gordon, James B Young, and Michael W Kattan. Individualizing risk prediction for positive coronavirus disease 2019 testing: results from 11,672 patients. *Chest*, 158(4):1364–1375, 2020.

Laure Wynants, Ben Van Calster, Gary S Collins, Richard D Riley, Georg Heinze, Ewoud Schuit, Marc MJ Bonten, Darren L Dahly, Johanna A Damen, Thomas PA Debray, et al. Prediction models for diagnosis and prognosis of covid-19: systematic review and critical appraisal. *bmj*, 369, 2020.

Dominik Janzing, Eleni Sgouritsa, Oliver Stegle, Jonas Peters, and Bernhard Schölkopf. Detecting low-complexity unobserved causes. *CoRR*, abs/1202.3737, 2012.

Peter Spirtes and Kun Zhang. Causal discovery and inference: concepts and recent methodological advances. In *Applied informatics*, volume 3, pages 1–28. SpringerOpen, 2016.

Christina Heinze-Deml, Marloes H Maathuis, and Nicolai Meinshausen. Causal structure learning. *Annual Review of Statistics and Its Application*, 5:371–391, 2018.

Daniel Malinsky and David Danks. Causal discovery algorithms: A practical guide. *Philosophy Compass*, 13(1): e12470, 2018.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in genetics*, 10:524, 2019.

Solon Barocas and Andrew D Selbst. Big data's disparate impact. *California law review*, pages 671–732, 2016.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. Fairness in machine learning. *Nips tutorial*, 1:2, 2017.

Stephan Bongers, Patrick Forré, Jonas Peters, and Joris M Mooij. Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49(5):2885–2915, 2021.

Christopher Frye, Colin Rowat, and Ilya Feige. Asymmetric shapley values: incorporating causal knowledge into model-agnostic explainability. *Advances in Neural Information Processing Systems*, 33:1229–1239, 2020.

Tom Heskes, Evi Sijben, Ioan Gabriel Bucur, and Tom Claassen. Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models. *Advances in neural information processing systems*, 33:4778–4789, 2020.

Jiaxuan Wang, Jenna Wiens, and Scott Lundberg. Shapley flow: A graph-based approach to interpreting model predictions. In *International Conference on Artificial Intelligence and Statistics*, pages 721–729. PMLR, 2021.

Qingyuan Zhao and Trevor Hastie. Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, 39(1):272–281, 2021.

Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding discrimination through causal reasoning. *Advances in neural information processing systems*, 30, 2017.

Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. *Advances in neural information processing systems*, 30, 2017.

Junzhe Zhang and Elias Bareinboim. Fairness in decision-making—the causal explanation formula. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. Issue: 1.

Karima Makhlouf, Sami Zhioua, and Catuscia Palamidessi. Survey on causal-based machine learning fairness notions. *arXiv preprint arXiv:2010.09553*, 2020.

Jennifer Strong. MIT Technology Review: Beating the AI hiring machines. `https://www.technologyreview.com/2021/08/04/1030513/podcast-beating-the-ai-hiring-machines/`, 2022. Accessed 2022-07-15.

Dan Geiger, Thomas Verma, and Judea Pearl. Identifying independence in bayesian networks. *Networks*, 20(5): 507–534, 1990.

Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. *Causation, prediction, and search*. MIT press, 2000.

Niklas Pfister, Evan G. Williams, Jonas Peters, Ruedi Aebersold, and Peter Bühlmann. Stabilizing variable selection and regression. *The Annals of Applied Statistics*, 15(3):1220 – 1246, 2021.

Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. The dangers of post-hoc interpretability: Unjustified counterfactual explanations. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, page 2801–2807, Macao, China, 2019. AAAI Press. ISBN 9780999241141.

Divyat Mahajan, Chenhao Tan, and Amit Sharma. Preserving causal constraints in counterfactual explanations for machine learning classifiers, 2020.

Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Thomas J Page Jr. Multivariate statistics: A vector space approach. *JMR, Journal of Marketing Research (pre-1986)*, 21 (000002):236, 1984.

Christopher M Bishop. Mixture density networks. Technical report, Aston University, 1994.

David M Bashtannyk and Rob J Hyndman. Bandwidth selection for kernel conditional density estimation. *Computational Statistics & Data Analysis*, 36(3):279–298, 2001.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28, 2015.

Brian L Trippe and Richard E Turner. Conditional density estimation with bayesian normalising flows. *arXiv preprint arXiv:1802.04908*, 2018.

Christina Winkler, Daniel Worrall, Emiel Hoogeboom, and Max Welling. Learning likelihoods with conditional normalizing flows. *arXiv preprint arXiv:1912.00042*, 2019.

Torsten Hothorn and Achim Zeileis. Predictive distribution modeling using transformation forests. *Journal of Computational and Graphical Statistics*, 30(4):1181–1196, 2021.

Rui Li, Michael TM Emmerich, Jeroen Eggermont, Thomas Bäck, Martin Schütz, Jouke Dijkstra, and Johan HC Reiber. Mixed integer evolution strategies for parameter optimization. *Evolutionary computation*, 21(1):29–64, 2013.

Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020. doi:10.1038/s41586-020-2649-2. URL `https://doi.org/10.1038/s41586-020-2649-2`.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL `http://github.com/google/jax`.

The pandas development team. pandas-dev/pandas: Pandas, February 2020. URL `https://doi.org/10.5281/zenodo.3509134`.

J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.

Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021. doi:10.21105/joss.03021. URL `https://doi.org/10.21105/joss.03021`.

Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*, 2018.

Du Phan, Neeraj Pradhan, and Martin Jankowiak. Composable effects for flexible and accelerated probabilistic programming in numpyro. *arXiv preprint arXiv:1912.11554*, 2019.

João Eduardo Montandon, Marco Tulio Valente, and Luciana L Silva. Mining the technical roles of github users. *Information and Software Technology*, 131:106485, 2021.

2.2 Paper II: Scientific Inference With Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena

Freiesleben, Timo, Gunnar König, Christoph Molnar, and Alvaro Tejero-Cantero. **Scientific Inference With Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena.** *arXiv preprint arXiv:2206.05487 (2022).*

Gunnar König contributed to the paper as co-author with significant contributions. Gunnar König wrote large parts of the section on causal learning. Gunnar König, Alvaro Tejero-Cantero, and Christoph Molnar added valuable new ideas, proofread and helped revise the paper. Timo Freiesleben wrote large parts of the paper and developed the initial idea. Alvara Tejero-Cantero helped design Figures 1,3,4, and 7 and contributed a paragraph on mechanistic models.

Scientific Inference With Interpretable Machine Learning

Analyzing Models to Learn About Real-World Phenomena

Timo Freiesleben
Munich Center for Mathematical Philosophy
& Graduate School of Systemic Neurosciences
LMU Munich

Gunnar König
Department of Statistics
LMU Munich & University of Vienna

Christoph Molnar
Independent Researcher
Munich

Alvaro Tejero-Cantero
Cluster of Excellence Machine Learning
New Perspectives for Science
University of Tübingen

Abstract

Interpretable machine learning (IML) is concerned with the behavior and the properties of machine learning models. Scientists, however, are only interested in models as a gateway to understanding phenomena. Our work aligns these two perspectives and shows how to design *IML property descriptors*. These descriptors are IML methods that provide insight not just into the model, but also into the properties of the phenomenon the model is designed to represent. We argue that IML is necessary for scientific inference with ML models because their elements do not individually represent phenomenon properties; instead, the model in its entirety does. However, current IML research often conflates two goals of model analysis — model audit and scientific inference – making it unclear which model interpretations can be used to learn about phenomena. Building on statistical decision theory, we show that IML property descriptors applied on a model provide access to relevant aspects of the joint probability distribution of the data. We identify what questions such descriptors can address, provide a guide to building appropriate descriptors and quantify their epistemic uncertainty.

Keywords: Scientific Modeling, Interpretable Machine Learning, Scientific Representation, Inference, XAI

1 Introduction

Scientists increasingly use machine learning (ML) in their daily work. This development is not limited to natural sciences like the geosciences (Reichstein et al. 2019) or material science (Schmidt et al. 2019), but also extends to social sciences such as education science (Luan and Tsai 2021) and archaeology (Bickler 2021).

When building predictive models for problems with complex data structures, ML outcompetes classical statistical models in both performance and convenience. Impressive recent examples of successful prediction models in science include the automated particle tracking at CERN (Farrell et al. 2018), or DeepMind's AlphaFold, which has essentially solved the protein structure prediction challenge CASP (Senior et al. 2020). In such examples, some see a paradigm shift towards theory-free science that "lets the data speak" (Kitchin 2014, Anderson 2008, Mayer-Schönberger and Cukier 2013, Spinney 2022). Indeed, prediction is one of the core aims of science (Luk 2017, Douglas 2009), but so are, as philosophers of science and statisticians emphasize, explanation and knowledge generation (Salmon 1979, Longino 2018, Shmueli et al. 2010). Focusing exclusively on prediction may therefore represent a historical step back (Toulmin 1961, Pearl 2018).

What hinders scientists from using ML models to gain real-world insights is model complexity and an unclear connection between model and phenomenon — the so-called *opacity problem* (Boge 2022, Sullivan 2020). Interpretable machine learning (IML, also called XAI, for eXplainable artificial intelligence) aims to solve the opacity problem by analyzing individual model elements or inspecting specific model properties (Molnar 2020). Different stakeholders with different goals hold diverse expectations of IML (Zednik 2021), including scientists (Roscher et al. 2020), ML engineers (Bhatt et al. 2020), regulatory bodies (Wachter et al. 2017), and laypeople (Arrieta et al. 2020). Due to this plurality, IML has been criticized for lacking a proper definition (Lipton 2018).

Nevertheless, scientists increasingly use IML for inferring which features are predictive of e.g. crop yield (Shahhosseini et al. 2020, Zhang et al. 2019), personality traits (Stachl et al. 2020), or seasonal precipitation (Gibson et al. 2021). Although researchers are aware that their IML analyses remain just model descriptions, it is often implied that the explanations, associations, or effects found also extend to the corresponding real-world properties. Unfortunately, drawing inferences with IML can currently be epistemically problematic because the interpretation methods are not designed for that purpose (Molnar et al. 2022). In particular, the difference between model-only versus phenomenon explanations is often unclear (Chen et al. 2020, Hooker et al. 2021), and a theory to quantify the uncertainty of interpretations is lacking (Molnar et al. 2020a, Watson 2022).

Contributions. In this paper, we present an account of scientific inference with IML inspired by ideas from philosophy of science and statistical inference. While we focus on supervised learning on identically and independently distributed (i.i.d.) data, we briefly discuss other learning scenarios in Section 5.3. Our key contributions are: 1. We argue that ML cannot profit from the traditional approach to scientific inference via model elements because its parameters do not represent phenomenon properties (Section 3). While current IML methods aim to restore representationality of the model as a whole, they conflate the model audit and scientific inference goals of interpretation. 2.

We identify the properties that IML methods need to fulfill to provide access to aspects of the conditional probability distribution $\mathbb{P}(\boldsymbol{Y} \mid \boldsymbol{X})$, where $\boldsymbol{X}$ describes predictor variables and $\boldsymbol{Y}$ the target (Section 4). We call methods that are suitable for inference *IML property descriptors*. We provide a guide to build such descriptors starting with a phenomenon question about $\boldsymbol{X}$ and $\boldsymbol{Y}$ and evaluating whether it can be addressed, followed by an answer to this question with ML models and finite data, and conclude with the quantification of epistemic uncertainty. We illustrate our approach using conditional partial dependence plots (cPDP) as an example IML descriptor.

Terminology. For the purposes of our discussion below, a *phenomenon* is a real-world process whose aspects of interest can be described by random variables. Observations of the phenomenon are drawn from the unknown joint distribution induced by the random variables and form the dataset or just *data*. A *ML model* is a mathematical model optimized with the aid of a learning algorithm applied on the collected data in order to accurately predict unknown or withheld phenomenon observations, i.e. to generalize beyond the initial data. Here we focus on the supervised learning setting. Finally, *scientific inference* is the process of rationally deriving conclusions about a phenomenon from data (via ML, or other types of models). We employ *inference* to imply investigating unobserved variables and parameters similar to statistical inference, i.e. in a more general sense than is common in some of the ML literature, where it is used exclusively as a synonym for prediction. The knowledge gained by scientific inference can build the basis of *scientific explanations*. These brief conceptual remarks are meant to reduce ambiguity in our usage: we lay no claim as to their universality.

2 Related Work

Whether and how ML models, and specifically IML, can help obtain knowledge about the world is a debated topic among philosophers of science, statisticians, and also the IML community.

Philosophy of Science. It has been argued that ML models are only suitable for prediction because their parameters are instrumental and lack meaning (Bailer-Jones and Bailer-Jones 2002, Bokulich 2011). On the other hand, Sullivan (2020) argues that nothing prevents us from gaining real-world knowledge with ML models as long as the *link uncertainty* — the connection between the phenomenon and the model — can be assessed. Cichy and Kaiser (2019) and Zednik and Boelsen (2022) claim that IML can help in learning about the real world, but they remain vague about how model and phenomenon are connected. Like Watson (2022), we explain that IML methods relying on conditional sampling are faithful to the phenomenon. However, while he assigns IML inferences to the causal phenomenon level, we clarify that, without additional assumptions, such inferences only reveal associational relationships (Räz 2022). Our work makes precise that ML models can be described as epistemic representations of a certain phenomenon that allow us to perform valid inferences (Contessa 2007) via interpretations.

Statistical Modeling and Machine Learning. Breiman et al. (2001) describes ML (algorithmic modeling) and statistics (data modeling) as two approaches for reaching conclusions from data. On a medical example he shows that post-hoc analysis of ML models can allow more correct inferences about the underlying phenomenon than standard, inherently interpretable data models. Our paper gives an epistemic foundation for such post-hoc analyses. Shmueli et al. (2010) distinguishes statistics and ML by their goals — prediction (ML) and explanation (statistics). Like Hooker and Mentch (2021), we argue against such a clear distinction and offer steps to integrate the two fields.

This paper builds on ideas from Molnar et al. (2021), where they introduce ground-truth and confidence intervals for partial dependence plots (PDP) and permutation feature importance (PFI) of arbitrary ML models. Our work generalizes these ideas to arbitrary IML methods and draws the connection to the underlying phenomenon.

Interpretable Machine Learning. IML as a field has been widely criticized for being ill-defined, mixing different goals (e.g. transparency and causality), conflating several notions (e.g. simulatability and decomposability), and lacking a proper measure of success (Doshi-Velez and Kim 2017, Lipton 2018). Some even argued against the central IML leitmotif of analyzing trained ML models post hoc in order to explain them (Rudin 2019). In this paper, we show that, if we focus on interpretations for scientific inference, a clear foundation including a proper theory of success can be provided and these criticisms can be partially addressed.

3 Scientific Inference and Elementwise Representationality

The goal of this paper is to analyze and describe how we can conduct scientific inference on ML models using IML methods. This section explains why inference with ML models cannot be done as in traditional scientific models and why current IML methods do not generally address the problem. The next section describes our solution and illustrates it with a complete example from question formulation to uncertainty quantification.

3.1 ML Models are not Elementwise Representational

In scientific modeling, there is a paradigm that many models implicitly follow — we call it the paradigm of *elementwise representationality*.

Definition. *A model is* elementwise representational *(ER) if all model elements (variables, relations, and parameters) represent an element in the phenomenon (components, dependencies, properties).*

Figure 1 depicts the relationship between ER models and the phenomenon:[1] variables describe phenomenon components; mathematical relations between variables describe structural, causal or associational dependencies between components; parameters specify the mathematical relations and describe properties of the component dependencies. The upward arrows describe *encoding* i.e. the translation of a phenomenon observation to a model configuration; The

[1] See Appendix A for the philosophical origins of our perspective.

downward arrows describe *decoding* i.e. the translation of knowledge about the model into knowledge about the phenomenon. ER is obtained through model construction; ER models are usually "hand-crafted" based on background knowledge and an underlying scientific theory. Variables are selected carefully and sparsely during model construction, and the relations are constrained to a relation class with few free parameters. When ER models need to account for an additional phenomenon aspect, they are gradually extended so that large parts of the "old" model are preserved in the more expressive "new" model. ER even eases this model extension process because model interventions are intelligible on the level of model elements. Usually, ER is explicitly enforced in modeling: if there is a phenomenon element devoid of meaning, researchers either try to interpret it or exclude it from the model.

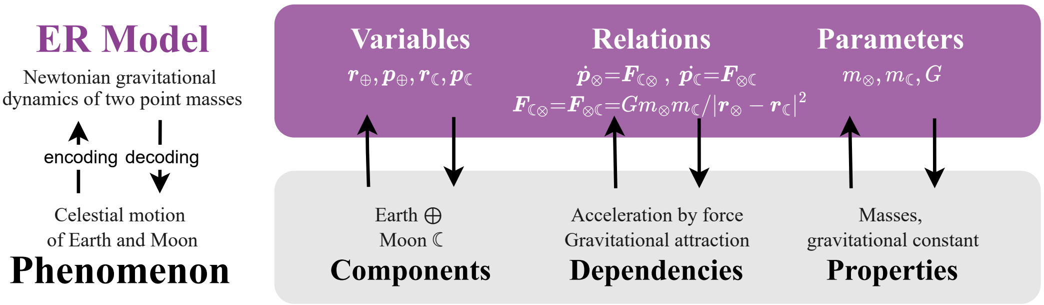

ER Model
Newtonian gravitational dynamics of two point masses
encoding
decoding
Celestial motion of Earth and Moon
Phenomenon
Variables
$\boldsymbol{r}_\oplus, \boldsymbol{p}_\oplus, \boldsymbol{r}_☾, \boldsymbol{p}_☾$
Relations
$\dot{\boldsymbol{p}}_\otimes = \boldsymbol{F}_{☾\otimes}$, $\dot{\boldsymbol{p}}_☾ = \boldsymbol{F}_{\otimes☾}$
$\boldsymbol{F}_{☾\otimes} = \boldsymbol{F}_{\otimes☾} = G m_\otimes m_☾ / |\boldsymbol{r}_\otimes - \boldsymbol{r}_☾|^2$
Parameters
$m_\otimes, m_☾, G$
Earth ⊕
Moon ☾
Components
Acceleration by force
Gravitational attraction
Dependencies
Masses, gravitational constant
Properties

Figure 1: **Model and phenomenon sustain an encoding-decoding relationship**. The main elements of a traditional, ER model, are shown in encoding-decoding correspondence to the phenomenon elements they represent (Stachowiak 1973). Phenomenon and model elements are illustrated with a simple example of two bodies in gravitational interaction and its classical, Newtonian mechanistic description.

ER is so remarkable because it gives models capabilities that go beyond prediction. ER simplifies the step of decoding i.e. translating model knowledge into phenomenon knowledge. Scientists can analyze model elements and draw immediate conclusions about the represented phenomenon element (Frigg and Nguyen 2021). However, only those aspects of the phenomenon that have a model counterpart can be analyzed with this approach. Fortunately, as described above, ER models can be extended to account for further relevant aspects identified by the scientist.

Running example:[2] *Linear Model.* Suppose a researcher, we call her Laura, wants to study what attributes influence students' grades in mathematics. Specifically, she wants to research how language skills and math skills are associated. She uses data from Cortez and Silva (2008), who collected a dataset[3], encompassing 32 student attributes in Portuguese schools including math/Portuguese grades, age, parents' education, etc.

Laura starts with a classical ER model — a linear model with one predictor and one target variable. She selects the student grade in Portuguese $\boldsymbol{X}_p$ and in mathematics $\boldsymbol{Y}$ as her proxy variables for students language and math skills respectively.[4] Based on her background knowledge, she assumes that the true relationship can be described as $\boldsymbol{Y} = \beta_0 + \beta_1 \boldsymbol{X}_p + \epsilon$ with $\beta_0, \beta_1 \in \mathbb{R}$ and an error $\epsilon \sim \mathcal{N}(\mu, \sigma^2)$. Laura centers $\boldsymbol{X}_p$ by the average student grade in

[2] Since the physical model from Figure 1 is a mechanistic causal model (Schölkopf et al. 2021), we switch henceforth to an illustrative associational model from the social sciences that compares more fairly with current associational ML models. We strongly simplify things in this example and do not claim that it reflects social science methodology or that ML is even required.

[3] see Appendix B for more details.

[4] In the Portuguese grading scheme, the range is 0-20, where 0 is the worst and 20 the best grade.

Portuguese and obtains the prediction model that minimizes the mean-squared-error (MSE),

$$\hat{m}_{\text{LIN}}(x_p) = 10.46 + 0.77x_p.$$

Laura's model is ER: she can interpret $\hat{\beta}_0 = 10.46$ as the predicted math grade for an average Portuguese student (if $x_p = 12.55$)[5] and $\hat{\beta}_1 = 0.77$ as the strength of association between the Portuguese grade and the math grade.

Laura can analyze the model to draw scientific inferences about the underlying phenomenon, for example, with 95% confidence intervals[6] for her estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ with $[10.05; 10.88]$ and $[0.63; 0.91]$ respectively. The inference she draws is on these parameters; Laura can only draw conclusions about the phenomenon if the model is ER and highly predictive. Laura may conclude from $\hat{\beta}_1$ that language skills and math skills are strongly and positively related. To reach a more expressive and predictively accurate model, Laura can also extend the model to include additional features, relations, or interaction terms. As long as she preserves ER, she can directly draw scientific inferences from analyzing model elements. Indeed, these inferences are only as valid as the modeling assumptions (e.g. target normality, homoscedasticity, or linearity).

ML Models are generally not ER. ER makes model elements interpretable and allows to reason about the effects of model or even real-world interventions; as such, ER models suit our image of science as an endeavor aimed at understanding. However, as mentioned above, for ER we usually require background knowledge on which components are relevant, and we need to severely restrict the class of relations that can be considered for the given phenomenon. These difficulties might lead scientists to either limit their investigations to phenomena that are already well studied or, as Breiman et al. (2001) argued, to develop overly simple models for complex phenomena and possibly draw wrong conclusions.

ML models excel for complex problems with an unbounded number of components that display ambiguous and entangled relationships i.e. ML models are highly expressive (Gühring et al. 2020). ML models are subject-domain independent (Bailer-Jones and Bailer-Jones 2002), this means that we do not necessarily need subject-domain background knowledge in modeling. Instead, ML modeling only requires specifying a broad model class and a set of hyperparameters. The choice of these hyperparameters is data-domain specific i.e. they reflect inductive biases that allow for efficient learning.

The gain in generality and convenience with ML comes at a price — ML models are generally not ER. As also argued in Boge (2022), Bokulich (2011), Bailer-Jones and Bailer-Jones (2002), ML models (e.g. artificial neural networks) contain model elements such as weights, activation functions, or network structure that have no corresponding phenomenon counterpart.

[5] Centering features is common in linear regression to make the intercept term interpretable.

[6] i.e., intervals $[a, b]$ such that, if the model assumptions hold, a 'true' parameter β is found inside 95% of all observational samples, $\mathbb{P}(a<\beta<b) = 0.95$.

Running example: Artificial Neural Network (ANN). Suppose Laura is dissatisfied with her linear model and fits a dense three-layer neural network to predict math grades using all available features.[7] She reduces the test-set MSE from 16.0 in the linear-one-variable model case to 8.9. A formal description of the model is given by:

$$\hat{m}_{\text{ANN}}(\boldsymbol{x}) = \sigma_3(W_3\sigma_2(W_2\sigma_1(W_1\boldsymbol{x} + b_1) + b_2) + b_3)$$

where model elements are the values of the weight matrices W_1, W_2, W_3 and bias vectors b_1, b_2, b_3, and the activation functions $\sigma_1, \sigma_2, \sigma_3$. Unlike in the linear model above, it is highly unclear what these parameters correspond to in our data or phenomenon. While the input vector x is still representational, the weights, activation functions, or three-layer architecture are very hard or even impossible to interpret: A high value of weight $W_1^{(3,2)}$ might have a positive, neutral, or negative effect on the target, dependent on all other model elements; the activation function only reflects the currently popular heuristics in model training; and the particular three-layer architecture is a result of model selection based on predictive performance and rules of thumb, but with little phenomenon-based rationale.

3.2 Scientific Inference in Light of Current IML

We have argued so far that:

i) If models are ER, they allow for scientific inference.

ii) ML models are generally not ER.

How can we still do scientific inference with ML models? We discuss two strategies to enable scientific inference with ML: We argue that the first strategy, namely restoring ER, fails because ML models are designed to represent in a distributed manner; the second strategy, embracing holistic representationality, is highly promising but current attempts conflate different goals of model analysis. This discussion sets the stage for the next section, where we show how a holistic account of representationality can enable scientific inference.

Restore ER. One strategy towards scientific inference with ML is to challenge Proposition ii) and show that ML models are ER too. Researchers in this camp argue that individual elements in ML have a natural phenomenon counterpart, but this counterpart only becomes evident when these model elements are extensively scrutinized.[8] This would be surprising: ER is not enforced in state-of-the-art techniques and, even worse, some methods such as training with dropout purposefully discourage ER in order to gain robustness (Srivastava et al. 2014); ML models like ANNs are designed for *distributed representation* (Buckner and Garson 2019, McClelland et al. 1987).

It has been claimed that model elements represent high-level constructs constituted from low-level phenomenon

[7] We chose a neural net to make our argument. For training the neural network, Laura splits data into training and test, uses ReLu activation functions and minimizes the MSE loss via gradient descent with an adaptive learning rate.

[8] The underlying epistemological reasoning is that human representations are near-optimal and will be eventually rediscovered by ML algorithms.

components that are often called *concepts* (Buckner 2018, Olah et al. 2020).[9] If this is the case, model elements or aggregates of such elements can be reconnected to the phenomenon; ER would be restored by the representations of coarse-grained phenomenon components. Research on neural networks supports that some model elements are associated with concepts (Mu and Andreas 2020, Voss et al. 2021, Kim et al. 2018, Olah et al. 2017), however, often these elements are neither the only associated elements nor exclusively associated with one concept as shown in Figure 2 (Donnelly and Roegiest 2019, Bau et al. 2017, Olah et al. 2020). Problematically, intervening on these model elements generally does not have the expected effect on the prediction — the elements do not share the causal role of the "represented" concepts, even in prediction (Gale et al. 2020, Donnelly and Roegiest 2019). It is therefore questionable in what sense they still represent.[10] Moreover, this line of research predominantly focuses on images, where nested concepts are arguably easier to identify for humans.

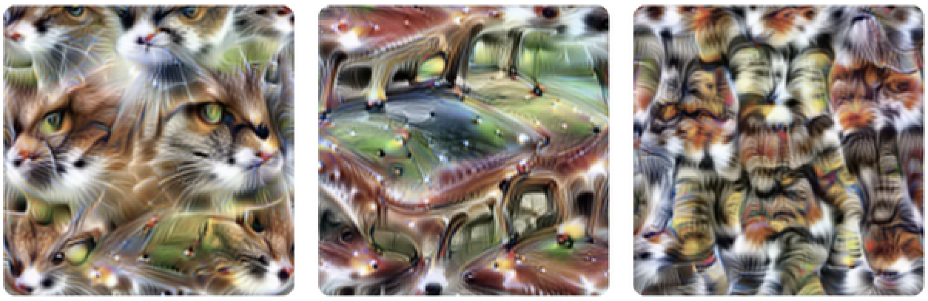


Figure 2: **ML models are generally not ER**. Three input images that independently trigger a single model element (unit 55 in layer *mixed4e*, Olah et al. (2017; 2020)). A single unit in a neural net may respond to very different "concepts", e.g. heads of cats (left image), car bodies (center), or bees (right), suggesting that units generally do not represent disentangled concepts (Mu and Andreas 2020, Nguyen et al. 2016).

Research on the representational correlates of model elements seems indeed fascinating. However, current ML models that do not enforce ER will rely on distributed representations and cannot be reduced to logical concept machines. The associative connection between model elements and phenomenon concepts should not be confused with their equivalence. Analyzing single model elements will therefore be a hopeless enterprise.

Embrace Holistic Representationality. An alternative route to scientific inference is to accept that ER is well-suited for scientific inference and that ML models are not ER but reject that ER is the only approach for scientific inference. To choose this route, one must offer an alternative path for drawing scientific inference with ML models that goes beyond the analysis of model elements.

Our approach is to regard the model as representational of phenomenon aspects *only as a whole* — we call this *holistic representationality* (HR). HR implicitly underlies large parts of the current research program in IML: Model-agnostic methods, in particular, analyze the entire ML model simply as an input-output mapping (Scholbeck et al.

[9] The idea is that similar to the hierarchical structure of components in nature, where lower level components such as atoms combine to form higher level entities such as molecules, cells, and organisms; in deep nets, hierarchies evolve from pixels to shapes to objects.

[10] Though note generative adversarial networks as an exception; here, interventions on model elements have been linked to interventions on concepts in the generated images (Bau et al. 2018).

2019); In the same spirit, many model-specific IML methods like gradient or path-based feature attribution treat ML models as mappings with additional useful properties such as differentiability (Alqaraawi et al. 2020).

Model-agnostic and model-specific methods share the idea that relevant model properties such as the effects or importances of variables can be derived by analyzing the model just as a functional mapping. Initial definitions of, for example, global feature effects (Friedman et al. 1991) and feature importance (Breiman 2001) or local feature contribution (Štrumbelj and Kononenko 2014) and model behavior (Ribeiro et al. 2016) have been presented. However, many researchers have pointed out that these methods lead to counterintuitive results for dependent or interacting features and offered alternative definitions (Apley and Zhu 2020, Strobl et al. 2008, Molnar et al. 2020b, Goldstein et al. 2015, König et al. 2021b, Janzing et al. 2020, Slack et al. 2020, Alqaraawi et al. 2020).

We believe that these controversies stem from a lack of clarity about the goal of model analysis. Are we interested in model properties to learn about the model (model audit) or do we want to use these model properties as a gateway to learn about the underlying phenomenon (scientific inference)? These two goals must not be conflated.

The auditor examines model properties e.g. for debugging, to check if the model satisfies legal or ethical norms, or to improve her understanding of the model by intervening on it (Raji et al. 2020). Auditors even take interest in model properties that have no corresponding phenomenon counterpart such as single model elements or the model behavior for unrealistic feature combinations. The scientist who wants to draw inferences, on the other hand, wants to learn about model properties that can be interpreted in terms of the phenomenon.

Scientific inference and model audit should be viewed as two different but interacting goals. In each of them, we take different stances toward the ML model: The auditor adopts a skeptical attitude of the model, she has ground-truth information or normative standards to check the model against; the scientist adopts a trusting attitude, she wants to learn from the model. Both cases describe a knowledge asymmetry (Gobet 2018, Rosser et al. 2008) but in opposite directions. Auditing the model is an indispensable step for scientists to gain enough trust in it. Only after several rounds of auditing and improvement should the researcher rely on the model to draw scientific conclusions.

4 Scientific Inference with IML Property Descriptors

We just argued that ML models are generally not ER and therefore do not allow for scientific inference in the standard way. HR offers a viable alternative, but currently different goals of model analysis are conflated. In this section, we show that a HR perspective enables scientific inference using IML methods. Particularly, we show that certain IML methods — we call them *IML property descriptors* — can represent phenomenon properties. Figure 3 describes our conceptual move: instead of matching phenomenon properties with model parameters as in ER models, we match them with external descriptions of the whole model.

Idea. Instead of first thinking about the model and its properties (the model audit approach), we propose to start with the phenomenon and a scientific question about it. IML methods for inference should answer, or at least help answer, a scientific question concerning the phenomenon. The crucial step in our framework is to establish a link between

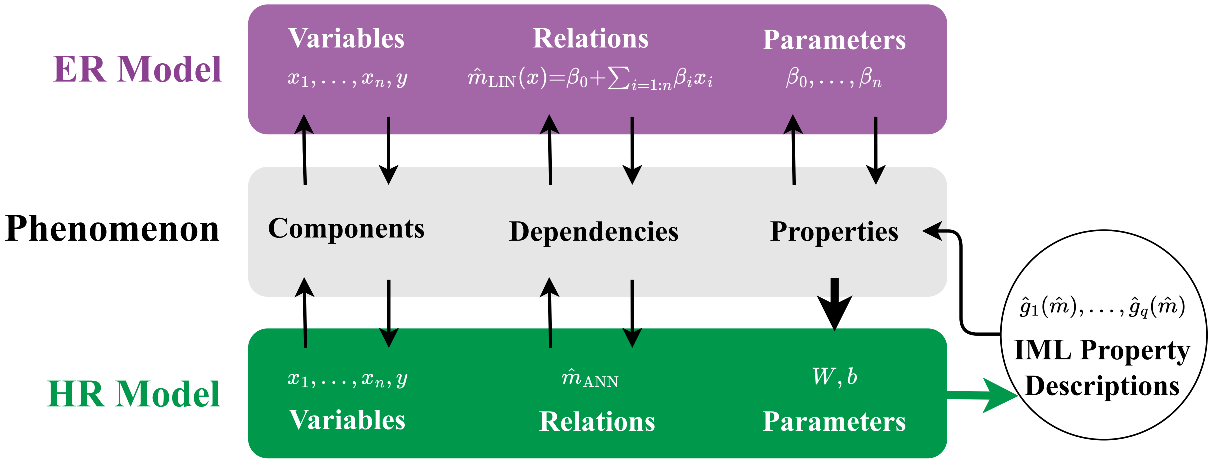


Figure 3: **IML property descriptions distill phenomenon properties from HR models**. Instead of explicitly encoding phenomenon properties as parameters like for ER models, HR models (e.g. ML models) encode phenomenon properties in the whole model. We propose that these encoded properties can be read out with IML property descriptions external to the model. In this way, IML offers an indirect route to scientific inferences through model analysis.

the phenomenon and the model; we propose to draw this link using statistical decision theory, which shows what optimal ML models can holistically represent. Clearly, an approximate ML model will not provide an answer if even the optimal model cannot. However, if a question can in principle be answered with the optimal model, an ML model trained on data can approximate the answer in practice. The problem then becomes to quantify the approximation error.

4.1 ML Representationality and Optimal Predictors

Which aspects of a phenomenon ML models can represent even under ideal circumstances depends on the data, the learning paradigm, and the loss function. For identically and independently distributed (i.i.d.) data used for supervised learning, optimal predictors from statistical decision theory provide an answer (Hastie et al. 2009, p18-22). Besides the advanced theory available in this setting, supervised learning on i.i.d. data is the most popular ML setup in practical applications. We briefly discuss representationality and scientific inference in the case of unsupervised and causal learning in Section 5.3.

Basic Notation. We assume that the random variables $X_1, \ldots, X_n$ and Y fully characterize the phenomenon. We write the joint feature vector as $X := (X_1, \ldots, X_n)$ with $\mathcal{X} := \text{Range}(X)$ and $\mathcal{Y} := \text{Range}(Y)$. X and Y jointly describe the phenomenon.

Optimal Predictors. An optimal predictor m can predict realizations of the target Y from realizations of X with minimal expected prediction error i.e. $m = \underset{\hat{m} \in \mathcal{M}}{\arg\min}\, \text{EPE}_{Y|X}(\hat{m})$, with $\text{EPE}_{Y|X}(\hat{m}) := \int_{Y} L(Y, \hat{m}(X))\, \mathbb{P}_{Y|X}(y|x)\, dy$, where

L describes a loss function $L(\boldsymbol{Y}, m(\boldsymbol{X})) : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^{+}$ and $\hat{m}$ a model in the set $\mathcal{M}$ of mappings from $\mathcal{X}$ to $\mathcal{Y}$. Table 1 shows the optimal predictors for standard loss functions.

Problem	**Loss**	$L(\boldsymbol{Y}, \hat{m}(\boldsymbol{X}))$	**Optimal Predictor** m
Regression	Mean Squared Error	$(\boldsymbol{Y} - \hat{m}(\boldsymbol{X}))^2$	$\mathbb{E}_{\boldsymbol{Y}\mid\boldsymbol{X}}[\boldsymbol{Y} \mid \boldsymbol{X}]$
	Mean Absolute Error	$\lvert\boldsymbol{Y} - \hat{m}(\boldsymbol{X})\rvert$	Median$(\boldsymbol{Y} \mid \boldsymbol{X})$
Classification	0-1 Loss	0 if $\hat{m}(\boldsymbol{X}) = \boldsymbol{Y}$, else 1	$\arg\max_{y\in\mathcal{Y}} \mathbb{P}(\boldsymbol{Y}{=}y \mid \boldsymbol{X})$
	KL divergence	$\sum_{r\in\mathcal{Y}} \mathbb{P}_{\boldsymbol{Y}}(r) \log\Big(\frac{\mathbb{P}_{\boldsymbol{Y}}(r)}{\mathbb{P}_{\hat{m}(\boldsymbol{X})}(r)}\Big)$[11]	$\mathbb{P}(\boldsymbol{Y} \mid \boldsymbol{X})$

Table 1: **The optimal predictors for standard loss functions can be derived from** $\mathbb{P}(\boldsymbol{Y} \mid \boldsymbol{X})$.

Supervised learning. Supervised learning seeks to find an optimal predictor m by using a learning algorithm[12] $I{:}\Delta \to \mathcal{M}$ that selects a a model $\hat{m}$ from a set $\mathcal{M}$ with the aid of a dataset $\mathcal{D} := ((x^{(1)}, y^{(1)}), \ldots, (x^{(k)}, y^{(k)}))$ with $\mathcal{D}$ in the set of datasets Δ drawn i.i.d. from the joint distribution, i.e. $(x^{(i)}, y^{(i)}) \sim (\boldsymbol{X}, \boldsymbol{Y})$. Instead of the EPE itself, the learning algorithm minimizes the empirical risk on the *test data* (i.e. on data not used to train $\hat{m}$), which is a finite-data estimate of the EPE.

4.2 IML Property Descriptors

We have just argued that ML models, when considered as a whole, approximate phenomenon aspects that can be derived from the conditional distribution $\mathbb{P}(\boldsymbol{Y} \mid \boldsymbol{X})$. *IML property descriptors* can help to investigate these aspects by describing their relevant properties.

Five Steps Towards IML Methods for Inference. Our proposal consists of the five steps in Figure 4, which we now discuss in detail. For each of the five steps, we provide an inference example based on the prediction of student grades in mathematics. In what follows, we assume that we have a supervised learning ML model $\hat{m}$ that approximates a phenomenon aspect described by the optimal predictor m.

Step 1) Formalize Scientific Question. Science starts by formulating a question. To address it with ML, this question has to be formalized. Exemplary questions that can already be addressed with IML methods following the scheme below are discussed in Section 4.3. Note that IML for scientific inference only helps answer questions that concern the association between $\boldsymbol{X}$ and $\boldsymbol{Y}$.

[11] This describes the forward KL divergence KL$(\boldsymbol{Y} \| \hat{m}(\boldsymbol{X}))$ for discrete $\boldsymbol{Y}$ and $\hat{m}(\boldsymbol{X})$, which differs from the backward KL divergence KL$(\hat{m}(\boldsymbol{X}) \| \boldsymbol{Y})$.

[12] The domain of I is only completely specified when the parameters that define the learning procedure and the search space of the algorithm (called hyperparameters in the context of $\hat{m}$) are fixed. For our discussion, the reader may assume hyperparameters to have been set a priori by a human or an automated ML algorithm (Hutter et al. 2019).

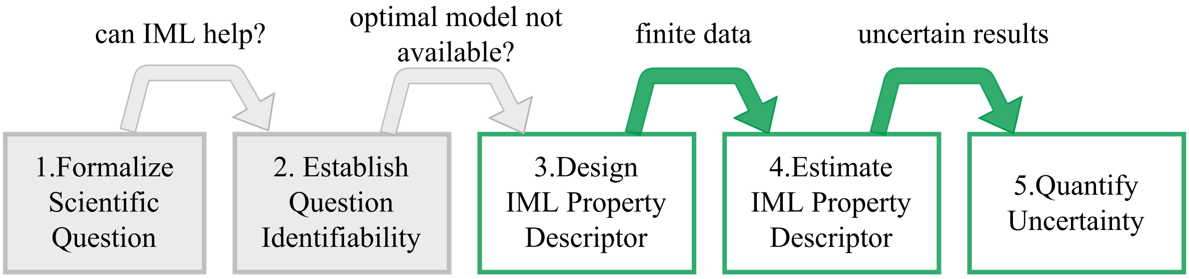

can IML help?
optimal model not available?
finite data
uncertain results
1.Formalize Scientific Question
2. Establish Question Identifiability
3.Design IML Property Descriptor
4.Estimate IML Property Descriptor
5.Quantify Uncertainty

Figure 4: **An epistemic foundation for scientific inference with IML**. Steps 1 and 2 clarify *what* kinds of scientific inferences we can draw with IML. Steps 3, 4, and 5 show *how* to draw such inferences and provide estimates of their precision.

Formally. We denote the formalized question by Q.

Example. Suppose Laura wants to find out how students' language skills are related to their math skills. She approaches this problem by asking how students' expected math grades are related to their Portuguese grades. Laura formalizes this question as the conditional expectation i.e. $Q = \mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X}_p}[\boldsymbol{Y} \mid \boldsymbol{X}_p]$, where $\boldsymbol{Y}, \boldsymbol{X}_p$ respectively stand for the math and Portuguese grade variable.[13]

Step 2) Establish Question Identifiability. Many scientific questions cannot be addressed using an ML model. ML models can only help answer questions that could theoretically be addressed with the optimal predictor. We call a question that the optimal predictor can help answer together with additional probabilistic knowledge (e.g. aspects of $\mathbb{P}(\boldsymbol{X}, \boldsymbol{Y})$) *identifiable*. A constructive strategy to establish identifiability relative to some probabilistic knowledge is to think of the transformations, using solely the probabilistic knowledge, that take the optimal predictor into the question Q. Of course, it is desirable to keep to a minimum the amount of probabilistic knowledge required to identify the question.

Formally. The optimal predictor is denoted by m. We say that a question is *identifiable* relative to probabilistic knowledge K if we can compute Q from m and K.

Example. Assume that Laura has trained her neural network, which, unlike the simple linear model presented above, takes into account *all* available features $\boldsymbol{X}$, to minimize the MSE loss, i.e. $m(\boldsymbol{X}) = \mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X}}[\boldsymbol{Y} \mid \boldsymbol{X}]$. Is Laura's question identifiable? For specific values of $\boldsymbol{X}$, the optimal predictor allows to compute the expected value of $\boldsymbol{Y}$ i.e. $m(x) = \mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X}}[\boldsymbol{Y} \mid \boldsymbol{X}=x]$. The only difference to Q is that m takes into account features besides the Portuguese grade, that we denote $\boldsymbol{X}_{-p}$. If we have access to the conditional distribution $\mathbb{P}(\boldsymbol{X}_{-p} \mid \boldsymbol{X}_p)$ (required[14] probabilistic knowledge

[13] This conditional expectation is the best possible point estimate of the math grade under the MSE loss, given just the Portuguese grades.

[14] Usually we do not have access to probabilistic knowledge K. We discuss this in more detail in Step 4.

K), we can integrate these other features out by taking the expected value

$$\begin{aligned} Q &:= \mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X}_p}[\boldsymbol{Y} \mid \boldsymbol{X}_p] \\ &= \mathbb{E}_{\boldsymbol{X}_{-p}|\boldsymbol{X}_p}[\mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X}}[\boldsymbol{Y} \mid \boldsymbol{X}] \mid \boldsymbol{X}_p] \qquad \text{(by the \textit{tower rule}, see App. C)} \\ &= \mathbb{E}_{\boldsymbol{X}_{-p}|\boldsymbol{X}_p}[m(\boldsymbol{X}) \mid \boldsymbol{X}_p]. \end{aligned}$$

Thus, Q is identifiable via m given $K = \mathbb{P}(\boldsymbol{X}_p \mid \boldsymbol{X}_{-p})$.

Step 3) Design IML Property Descriptor. It is not enough to identify a question. We need a way to estimate an answer for ML models — we need *IML property descriptors*. An IML property descriptor describes a continuous function that applies the transformation from the question identification step above to a given ML model and outputs an element of the space $\mathcal{Q}$. Thus, given the optimal predictor, an IML property descriptor outputs an answer to Q. Continuity guarantees that if our ML model is close to the optimal model, our answer is approximately correct. We call the application of a property descriptor to a specific ML model, $g_K(\hat{m})$, a *model property description.*

Formally. An *IML property descriptor* is a continuous function g_K (w.r.t. metrics $d_{\mathcal{M}}$ and $d_{\mathcal{Q}}$)[15] that identifies Q using probabilistic knowledge K:

$$g_K : \mathcal{M} \to \mathcal{Q} \quad \text{with} \quad g_K(m) = Q.$$

The output space $\mathcal{Q}$ remains unspecified to account for the variety of scientific questions; $\mathcal{Q}$ could denote a set of real numbers, vectors, functions, probability distributions, etc.

Example. The property descriptor describes the transformations that identify Q, i.e.

$$g_K(\hat{m})(x_p) := \mathbb{E}_{\boldsymbol{X}_{-p}|\boldsymbol{X}_p}[\hat{m}(\boldsymbol{X}) \mid \boldsymbol{X}_p{=}x_p]. \tag{4.1}$$

This is indeed a property descriptor because conditional expectation is continuous on $\mathcal{M}$, and Q is identifiable given $K = \mathbb{P}(\boldsymbol{X}_{-p} \mid \boldsymbol{X}_p)$. Note that Equation (4.1) describes the well-known conditional partial dependence plot, or cPDP, also known as M-plot (Molnar 2020, Apley and Zhu 2020).

Step 4) Estimate IML Property Descriptor. Often we lack access to relevant probabilistic knowledge K. Instead, we have a finite amount of data on which we can evaluate our ML mapping, which we call the *evaluation data.* It may bundle up our training and test data $\mathcal{D}$ (see Section 4.1), as well as additionally available (unlabeled) data, and artificially generated data. The *IML property description estimator* describes a way to estimate property descriptions with access only to the ML model plus the evaluation data.

[15] The function $d_{\mathcal{M}}$ is a metric on the function space $\mathcal{M}$, $d_{\mathcal{M}}(m_1, m_2) := \int_{\mathcal{X}} L(m_1(x), m_2(x))\, \mathbb{P}_X(x)\, \mathrm{d}x$ for $m_1, m_2 \in \mathcal{M}$, while $d_{\mathcal{Q}}$ describes a metric appropriate for the space $\mathcal{Q}$.

Formally. We denote the *evaluation dataset* by $\mathcal{D}^*$ and the random process that generates it by $\boldsymbol{D}^*$. We call $\hat{g}_{\mathcal{D}^*} : \mathcal{M} \to Q$ the *IML property description estimator* if it is an unbiased estimator of g_K i.e.

$$\mathbb{E}_{\boldsymbol{D}^*}[\hat{g}_{\boldsymbol{D}^*}(\hat{m})] = g_K(\hat{m}) \quad \text{for all } \hat{m} \in \mathcal{M}.$$

Example. Laura's evaluation dataset $\mathcal{D}^*$ is her initial training and test dataset $\mathcal{D}$ augmented by artificial instances created by the following manipulation: Laura makes six copies of the data, and jitters the Portuguese grade by $1, -1, 2, -2, 3$ or -3 respectively. This augmentation strategy reflects how Laura understands the Portuguese grade as noisy based on her background knowledge of how much student performance varies daily and teachers grade inconsistently. Let the students with (jittered) Portuguese grade i be $\mathcal{D}^*_{|x_p=i} := (x \in \mathcal{D}^* \mid x_p = i)$, then, we can define the IML property description estimator at i as the conditional mean (unbiased estimator of the conditional expectation):

$$\hat{g}_{\mathcal{D}^*}(\hat{m})(i) := \frac{1}{|\mathcal{D}^*_{|x_p=i}|} \sum_{x \in \mathcal{D}^*_{|x_p=i}} \hat{m}(x) \tag{4.2}$$

The estimated answer to Laura's question is plotted in Figure 5. The plot on the left suggests that math grade is only strongly dependent on Portuguese grades in the interval $8 - 17$. However, as we show in the next step, we must also take into account that we have very sparse data in some regions (e.g. very few students scored below 8) before confirming this first impression.

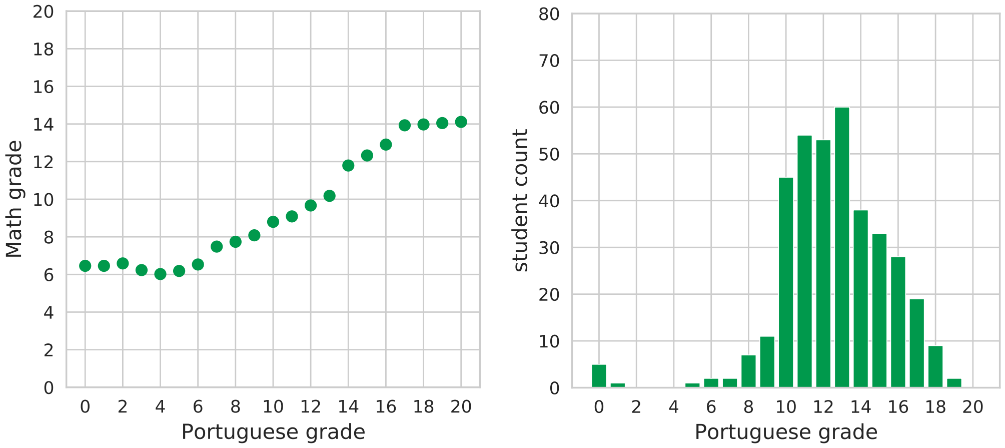


Figure 5: **Left Plot:** Estimate of $\mathbb{E}_{Y|X_p}[Y \mid X_p]$ via Equation (4.2). **Right Plot:** Histogram of grades in Portuguese.

Step 5) Quantify Uncertainty. We have shown how we can estimate Q using an approximate ML model paired with a suitable evaluation dataset. But how good is our estimate? Two steps involve approximations:

1. Applying the IML property descriptor to the ML model $\hat{m}$ instead of the optimal model m; we call the resulting error *model error*

$$\text{ME}[\hat{m}] = d_Q\big(g_K(m), g_K(\hat{m})\big).$$

The model error depends on the ML model $\hat{m}$ we obtained from the learning algorithm trained on the given dataset.

2. Applying the IML property description estimator on our evaluation dataset $\mathcal{D}^*$ instead of computing the true model property description based on K directly; we call this error *estimation error*

$$\mathrm{EE}[\mathcal{D}^*] = d_Q\Big(g_K(\hat{m}), \hat{g}_{\mathcal{D}^*}(\hat{m})\Big).$$

The estimation error depends on the evaluation dataset $\mathcal{D}^*$.

In theory, the model error and the estimation error can be separated. In practice, however, they are statistically dependent because the training and the evaluation data overlap. Generally, neither the model error nor the estimation error can be computed perfectly; this would require access to the optimal model m and infinitely many data instances. Nevertheless, we can quantify in expectation how large the two errors are.

An intuitive approach to quantifying the expected errors is to decompose them into bias and variance contributions. The two decompositions below quantify the range in which the true phenomenon property descriptions are most likely to lie.

Formally. For the bias-variance decomposition, we assume the metric d_Q to be the squared error.[16] Considering the dataset that we entered into the learning algorithm as a random variable $\boldsymbol{D}$, we can decompose the expected $\mathrm{ME}[\hat{m}]$ error as follows

$$\mathbb{E}_{\boldsymbol{D}}[\mathrm{ME}[\hat{m}]] = \underbrace{(g_K(m) - \mathbb{E}_{\boldsymbol{D}}[g_K(\hat{m})])^2}_{\text{Bias}^2} + \underbrace{\mathbb{V}_{\boldsymbol{D}}[g_K(\hat{m})]}_{\text{Variance}}$$

where $\hat{m} := I(\mathcal{D})$ is the output of a machine learning algorithm I for dataset $\mathcal{D}$ (Section 4.1). Considering the evaluation data as a random variable $\boldsymbol{D}^*$, we can decompose the expected EE_{D^*} error as follows

$$\mathbb{E}_{\boldsymbol{D}^*}[\mathrm{EE}[D^*]] = \underbrace{(g_K(\hat{m}) - \mathbb{E}_{\boldsymbol{D}^*}[\hat{g}_{\boldsymbol{D}^*}(\hat{m})])^2}_{\text{Bias}^2} + \underbrace{\mathbb{V}_{\boldsymbol{D}^*}[\hat{g}_{\boldsymbol{D}^*}(\hat{m})]}_{\text{Variance}} = \mathbb{V}_{\boldsymbol{D}^*}[\hat{g}_{\boldsymbol{D}^*}(\hat{m})].$$

The bias term vanishes because the property description estimator is by definition unbiased w.r.t. the IML property descriptor.

Example. Laura obtains different cPDPs (Figure 5) for different models with similar performance, as well as for different selections of evaluation data, how much can she then rely on these cPDPs?

The estimates of the variances of the cPDP by Molnar et al. (2021) allow to calculate pointwise confidence intervals (Figure 6). We can define a confidence interval that only incorporates the estimation uncertainty by

$$\mathrm{CI}_{\mathrm{EE}[\boldsymbol{D}^*]} := \Big[\hat{g}_{\mathcal{D}^*}(\hat{m})(i) \pm t_{1-\frac{\alpha}{2}} \sqrt{\hat{\mathbb{V}}_{\boldsymbol{D}^*}[\hat{g}_{\boldsymbol{D}^*}(\hat{m})(i)]}\Big]$$

[16] A bias-variance decomposition is also possible for other loss functions, including the 0-1 loss (Domingos 2000).

and a confidence interval that incorporates both model and estimation uncertainty by

$$\mathrm{CI}_{\mathrm{ME}[\hat{m}]\wedge\mathrm{EE}[\boldsymbol{D}^*]} := \left[\hat{g}_{\mathcal{D}^*}(\hat{m})(i) \pm t_{1-\frac{\alpha}{2}} \sqrt{\hat{\mathbb{V}}_{\boldsymbol{D},\boldsymbol{D}^*}[\hat{g}_{\boldsymbol{D}^*}(\hat{m})(i)]}\right].$$

For the combined confidence interval we require a strong and unfortunately not testable assumption to be satisfied — unbiasedness of the ML algorithm. Unbiasedness implies that, in expectation over training sets, the ML algorithm learns the optimal model, i.e. $m = \mathbb{E}_{\boldsymbol{D}}[\hat{m}]$.[17]

Figure 6 shows that for students with Portuguese grades between 8 and 17, Laura can be very confident in her model and the relationship it identifies between math and Portuguese grade.[18] However, both for Portuguese grades below 8 or above 17, the true value might be far off from our estimated value using a given model, as we can see from the width of the confidence intervals. For these grade ranges, gathering more data may reduce Laura's uncertainty.

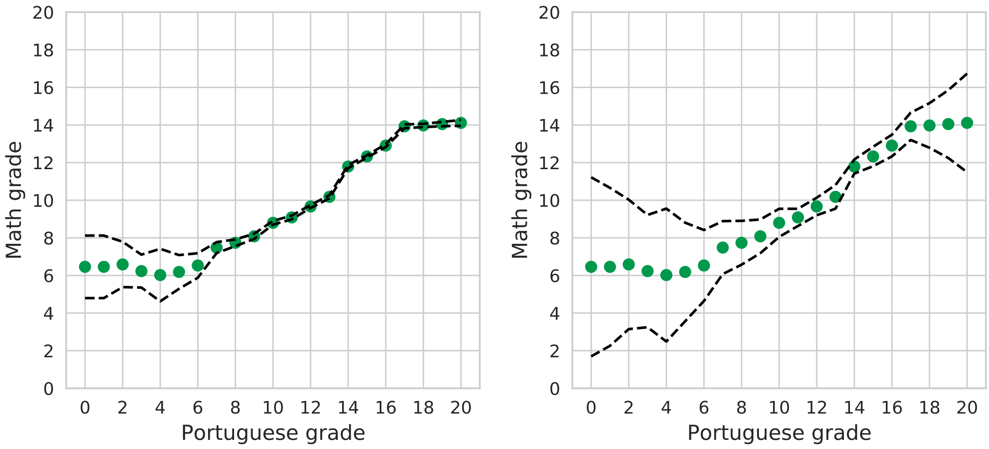


Figure 6: **Uncertainty evaluation of an IML property description**. **Left:** cPDP and its estimation error due to Monte-Carlo integration. **Right:** cPDP with *both* estimation and model error. Confidence bands in dashed lines cover the true expected math grade in 95% of all cases. These plots jointly suggest that most of the uncertainty is due to the model error.

Summary. Figure 7 gives an overview of all functions and spaces involved in IML for scientific inference. We started from a phenomenon and formalized a scientific question Q about it. Using a learning algorithm I on dataset $\mathcal{D}$ from the phenomenon, we learned an ML model $\hat{m}$ that approximates the optimal model m. We then set out to answer Q from $\hat{m}$. We defined a property descriptor g_K, that is, a function that allows to compute Q from m given K, respectively approximates Q from $\hat{m}$ given K. Because g_K requires probabilistic knowledge about $\mathbb{P}(\boldsymbol{X}, \boldsymbol{Y})$, we introduced a property description estimator $\hat{g}_{\mathcal{D}^*}$, a function estimating Q solely from finite data, the evaluation set $\mathcal{D}^*$. Finally, we showed how the expected error of our estimation steps can be quantified with confidence intervals $\mathrm{CI}_{\mathrm{ME}[\hat{m}]}$ and $\mathrm{CI}_{\mathrm{EE}[\boldsymbol{D}^*]}$.

[17] Since unbiasedness is tied to a specific context, there is no conflict with the no-free-lunch theorems (Sterkenburg and Grünwald 2021).

[18] We used resampling techniques to estimate the two variances. In real-data settings it is generally not possible to always sample new data for the model training and the evaluation. Although resampling may result in an underestimation of the variance, our goal here is simply to illustrate the process of quantifying uncertainty for a concrete IML method.

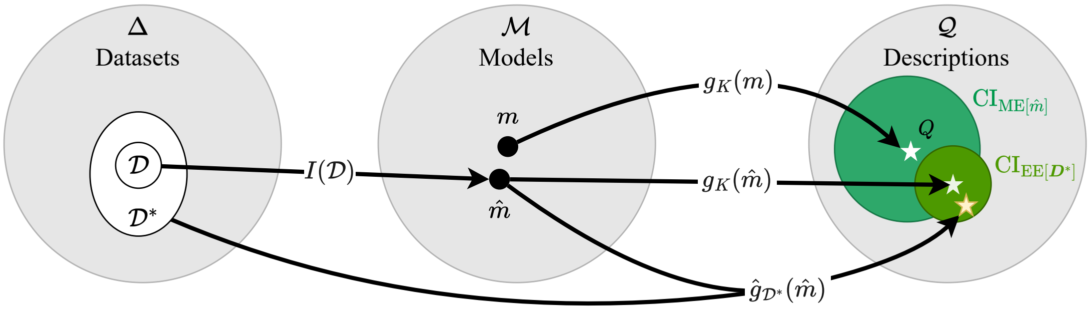


Figure 7: **From datasets to inferences via ML models**. Mappings are represented by arrows and sets are represented by filled circles, with confidence regions in green shades. Practical IML descriptions $\hat{g}_{\mathcal{D}^*}(\hat{m})$ are approximate, uncertainty-aware answers to a question Q that are built from a model $\hat{m}$ fit on $\mathcal{D}$ and an evaluation dataset $\mathcal{D}^*$.

4.3 Property Descriptors and Current IML Methods

Many questions that can be answered based on the conditional probability distribution $\mathbb{P}(\boldsymbol{Y} \mid \boldsymbol{X})$ are widely relevant. The goal of practical IML research for inference should be to define relevant descriptors and provide accessible implementations of these descriptors, including quantification of uncertainty. To find out which specific questions are relevant to scientists, and therefore what descriptors are necessary, IML researchers, statisticians and scientists must closely interact.

In Table 2 we present a few examples of elementary inference questions that can in principle be addressed by existing IML methods i.e. these methods can operate as property descriptors already. We distinguish between global and local phenomenon questions: global questions concern general associations, local questions concern associations for a specific instance. The last column highlights current IML methods that provide approximate answers, albeit often without uncertainty quantification. Note how we ultimately require conditional versions of existing marginal IML methods, which suggests that marginal sampling, which generates unrealistic instances, is inadequate in scientific inference.

5 Discussion

ER models enable straightforward scientific inference because their elements represent something about the underlying phenomenon. While ML models are generally not ER, IML can offer an indirect route to scientific inference, provided model properties have a corresponding phenomenon counterpart. We have shown how phenomenon representation can be achieved through optimal predictors and described how to practically construct IML property descriptors following five-steps: the first two steps clarify what questions we can address with IML, step three and four show how to answer them with ML models and finite data, and step five allows to evaluate how certain the answers are. We pointed out

[19] Only defined on phenomenon if $\mathbb{P}(X_p = p, X_{-p} = x_{-p}) > 0$.

[20] Only with the right similarity metric that accounts for the realistic constraint.

Global

Question	Formalization	IML method
Effect: What is the best estimate of $\boldsymbol{Y}$ if we only know $\boldsymbol{X}_p$?	$m_{\boldsymbol{X}_p}(\boldsymbol{X}_p)$	cPDP (Apley and Zhu 2020)
Conditional Contribution: How much worse can $\boldsymbol{Y}$ be predicted from $\boldsymbol{X}$ if we hadn't known $\boldsymbol{X}_p$?	$\mathrm{EPE}_{\boldsymbol{X},\boldsymbol{Y}}(m_{\boldsymbol{X}}(\boldsymbol{X})) - \mathrm{EPE}_{\boldsymbol{X}_{-p},\boldsymbol{Y}}(m_{\boldsymbol{X}_{-p}}(\boldsymbol{X}_{-p}))$	cPFI (Fisher et al. 2019)
Fair Contribution: What is the fair share of feature $\boldsymbol{X}_p$ in the prediction of $\boldsymbol{Y}$?	$\frac{1}{n} \sum_{S \subseteq \{1,\ldots,n\} \setminus j} \binom{n-1}{\lvert S \rvert}^{-1} \Big(\mathrm{EPE}_{\boldsymbol{X}_{S \cup \{j\}},\boldsymbol{Y}}(m_{\boldsymbol{X}_{S \cup \{j\}}}(\boldsymbol{X}_{S \cup \{j\}})) - \mathrm{EPE}_{\boldsymbol{X}_S,\boldsymbol{Y}}(m_{\boldsymbol{X}_S}(\boldsymbol{X}_S)) \Big)$	SAGE (Covert et al. 2020)
Relevant Value: Under which realistic conditions can we expect to observe relevant value y_{rel}?	$\underset{x \in \mathcal{X} \wedge \mathbb{P}(\boldsymbol{X}=x)>0}{\arg\min} \; d_{\mathcal{Y}}(m_{\boldsymbol{X}}(x), y_{rel})$	no method yet

Local

Effect: How does the best estimate of $\boldsymbol{Y}$ change relative to $\boldsymbol{X}_p$, knowing that $\boldsymbol{X}_{-p} = x_{-p}$?	$m_{\boldsymbol{X}}(\boldsymbol{X}_p, x_{-p})$	ICE-curve [19] (Goldstein et al. 2015)
Conditional Contribution: How much worse can $\boldsymbol{Y}$ be predicted from $\boldsymbol{X} = x$ if we hadn't known $\boldsymbol{X}_p$?	$L(y, m_{\boldsymbol{X}}(x)) - L(y, m_{\boldsymbol{X}_{-p}}(x_{-p}))$	no method yet
Fair Contribution: What is the fair share of feature $\boldsymbol{X}_p$ in the prediction of $\boldsymbol{Y}$ if $\boldsymbol{X} = x$?	$\frac{1}{n} \sum_{S \subseteq \{1,\ldots,n\} \setminus j} \binom{n-1}{\lvert S \rvert}^{-1} \Big(m_{\boldsymbol{X}_{S \cup \{j\}}}(x_S, x_j) - m_{\boldsymbol{X}_S}(x_S) \Big)$	conditional Shapley Values (Aas et al. 2021)
Relevant Value: Under which realistic conditions similar to $\boldsymbol{X} = x$ can we expect to observe relevant value y_{rel}?	$\underset{x' \in \mathcal{X} \wedge \mathbb{P}(\boldsymbol{X}=x')>0}{\arg\min} \; d_{\mathcal{Y}}(m_{\boldsymbol{X}}(x'), y_{\mathrm{rel}}) + \lambda \, d_{\mathcal{X}}(x, x')$	Counterfactuals[20] (Dandl et al. 2020)

Table 2: **Global and local formalized questions and matching IML property descriptors**. Note that questions are relative to a specific loss function L; for a set $S \subseteq \{1, \ldots, n\}$, the term $m_{\boldsymbol{X}_S}$ describes the optimal predictor of $\boldsymbol{Y}$ w.r.t. loss function L and random variable(s) $\boldsymbol{X}_i$ with $i \in S$. $d_{\mathcal{X}}$ and $d_{\mathcal{Y}}$ describe suitable metrics on $\mathcal{X}$ and $\mathcal{Y}$ respectively.

that some current IML methods can already be seen as IML property descriptors.

Is the lack of elementwise representationality specific to ML models? No, ML shows only an extreme case. In fact, there is a continuum between fully ER models and HR-only models: Some scientific models contain elements that are difficult or impossible to interpret e.g. the wave function in physics (Callender 2015); complex classical statistical models like generalized additive models also contain elements that are difficult to interpret. Our main message is: the five-step approach can be used to extend inference to any non ER model (whether ML or not).

One could argue that science should only rely on ER models (Rudin 2019). Indeed, it would be great if we could always build models from simple to complex and keep ER from beginning to end. However, more and more problems seem to be very difficult to tackle with this approach (Nearing et al. 2021); Interpretable but inaccurate models (w.r.t. to the phenomenon) are not a solution (Breiman et al. 2001). In situations where we cannot construct accurate ER models because we lack background knowledge or the phenomenon is very complex, scientific inference with ML models may thus be the only viable alternative.

5.1 Implications

Adopting a phenomenon-centric perspective on IML allows us to answer a variety of questions that were puzzling from a model-centric perspective:

Which questions can be addressed with IML property descriptors? IML property descriptors can help retrieve relevant phenomenon properties i.e. properties derived from the conditional distribution $\mathbb{P}(\boldsymbol{Y} \mid \boldsymbol{X})$. Which phenomenon properties are relevant is context-specific and up to researchers to identify. While formulating questions, researchers must be aware that supervised ML models are only representational of associative structure and not the underlying causal mechanism (see Section 5.3).

Why use (I)ML for inference? Supervised ML can help draw scientific inference when sampling from $\boldsymbol{X}$ is easy but sampling from $\boldsymbol{Y}$ is difficult, e.g. when $\boldsymbol{Y}$ is hard to measure or determined only in the future. In such situations, analyzing both the model and the data with IML methods can allow for better conclusions than analyzing just the data — the ML model fills the gaps by interpolation. Extrapolation to out-of-distribution data is generally not a strength of ML and can lead to incorrect conclusions; such extrapolations should only be trusted if the learning algorithm incorporated a powerful and suitable inductive bias.

When sampling from $\boldsymbol{X}$ is difficult or the property of interest can be computed more reliably by other means, we advice against using IML for inference.

How important is model performance in inference? If the model is a poor approximation or representation of the modeled phenomenon, the conclusions we draw from that model are unreliable (Cox 2006, Good and Hardin 2012). Thus, a good fit is vital for gaining reliable knowledge.

Note that even for the optimal model, there remains the so-called Bayes error rate, an irreducible error arising from the fact that $\boldsymbol{X}$ does not completely determine $\boldsymbol{Y}$ (Hastie et al. 2009). Thus, high error does not necessarily flag a low-quality model, but rather may indicate that $\boldsymbol{X}$ provides insufficient information about $\boldsymbol{Y}$.

What kind of data should be used for IML? Many IML methods (e.g. Shapley Values, LIME, etc.) rely on probing the ML model on permuted data (Scholbeck et al. 2019). These artificial "data" may never occur in the real world. This may be useful to audit the model, but if we want to learn about the world, artificial data is supposed to credibly supplement observations. Our analysis therefore substantiates the criticism of Hooker and Mentch (2021), Hooker et al. (2021), Mentch and Hooker (2016) concerning the permutation of features irrespective of the dependency structure in the data.

5.2 Open Problems

There are several open issues that we have not addressed:

What about non-tabular data? For some data types, such as images, audio, or video data, it is extremely difficult to formulate scientific questions only in terms of low-level features such as pixels or audio frequencies. To follow our approach, we need a translation of high-level concepts (e.g. objects in images or words in audio) that scientists can use to formulate their questions into low-level features (e.g. pixels or audio frequencies) that the model works with. Such translations are notoriously difficult to find; deep learning may help here (Jia et al. 2013, Zaeem and Komeili 2021, Zhou et al. 2018, Koh et al. 2020).

How to assess if data is realistic? In IML, we often need to augment our data. However, using unrealistic data is highly problematic for scientific inference, as mentioned earlier. Reasonable permutations of features such as Laura's grade jitter strategy (see Section 4.2), can supply realistic data. However, this requires expert knowledge about what permutations make sense. Conditional density estimation techniques or generative models (e.g. generative adversarial networks, normalizing flows, variational autoencoders, etc.) may provide additional paths to obtain realistic data. However, modeling the conditional density can be computationally intensive and more difficult than the original prediction problem, or may even be epistemically problematic since it only approximates sampling real data.

To what extent does a property determine the true model? Sometimes, we know that the model answer to a scientific question is correct. How strongly does this confirm the correctness of the model? Property descriptions narrow down the potential models and sufficiently many property descriptions can even completely determine the model, e.g. for the FANOVA decomposition (Apley and Zhu 2020, Hooker 2004). Model property descriptions may eventually be used to incorporate background knowledge in training. Both directions, extracting knowledge from ML models, and using background knowledge to build more adequate ML models, are elementary for scientific progress (Dwivedi et al. 2021, Nearing et al. 2021, Razavi 2021).

5.3 Other Forms of Scientific Inference With ML

In this paper, we focused exclusively on scientific inference with supervised learning ML models on i.i.d. data. For this setting, there is sufficient theory in both statistical decision theory and IML research to provide secure epistemic foundations for scientific inference. We have explained what we can learn about the conditional distribution of $\boldsymbol{Y}$ given $\boldsymbol{X}$. We can even learn that $\boldsymbol{X}$ contains little information about $\boldsymbol{Y}$ to predict it, which is scientifically interesting (Taleb 2005, Shmueli et al. 2010). However, many questions that scientists regularly face are of a different nature and go beyond conditional distributions.

Unsupervised Learning. Unsupervised learning is concerned with estimating aspects of the joint distribution $\mathbb{P}(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n)$. Unsupervised learning is hard as it typically targets a high-dimensional joint distribution and, often, lacks a clear measure of success (Hastie et al. 2009, p486). In principle, our five-step guide is also applicable to unsupervised learning, however, we lack a theoretical counterpart to optimal predictors.

Causal Learning. The observational joint probability distribution is interesting, but it remains on rung one of Judea Pearl's ladder of causation — the associational level (Pearl and Mackenzie 2018). What scientists are often much more interested in is answering causal questions such as average treatment effects (rung 2) or counterfactual questions (rung 3) (Salmon 1998, Woodward and Ross 2021). Laura may be interested not only whether students' language and math skills are associated (rung 1), but also in whether the provision of tutoring in Portuguese affects students' math skills (rung 2) or whether a specific student (who is not a native Portuguese speaker) would have done better in math if she had received Portuguese tutoring at a young age (rung 3).

Supervised ML models only represent aspects of the observational distribution (rung 1) and therefore generally do not allow answering causal questions. As a consequence, the IML descriptors of the models also generally do not allow causal insight into the data. Many IML papers that discuss causality (Schwab and Karlen 2019, Janzing et al. 2020, Wang et al. 2021, Heskes et al. 2020) are only concerned with causal effects on the model's prediction, which do not necessarily translate into a causal insight into the phenomenon.

In order to answer causal questions, causal models should be used instead.[21] To learn a causal model, we must gather interventional data and/or make strong, untestable assumptions. Causal inference constitutes thus a challenging problem and remains an active area of research (Heinze-Deml et al. 2018, Kalisch and Bühlmann 2014, Constantinou et al. 2021, Peters et al. 2017).

In certain situations, ML models can nevertheless be useful for causal inference. Firstly, if all predictor variables are causally independent and the prediction target is caused by the features, the causal model interpretation implies the causal data interpretation. Secondly, associative models in combination with IML can help estimate causal effects even in the absence of causal independence if they are in principle identifiable by observation. For example, the partial dependence plot coincides with the so-called adjustment formula and therefore identifies a causal effect if the backdoor criterion is met (and the model optimally predicts the conditional expectation) (Zhao and Hastie 2021). Thirdly, when there is access to observational and interventional data during training, training ML models with invariant risk minimization yields models that predict accurately in interventional environments (Peters et al. 2016, Pfister et al. 2021, Arjovsky et al. 2019). For such intervention-stable models, IML methods that provide insight into the effect of interventions on the prediction also describe causal effects on the underlying real-world components (König et al. 2021a).

Another way in which ML supports causal inference is by facilitating practical scientific inference relying on complex mechanistical models, frequently implemented as numerical simulators. Indeed simulators can represent complex, causal, dynamics in an ER fashion, but often at the price of an intractable likelihood and thus expensive inference. A variety of new ML methods for likelihood-free inference on simulators (Cranmer et al. 2020) allows to estimate a full posterior distribution over ER parameters for increasingly complex models.

While supervised learning learns from a fixed dataset, reinforcement learning (RL) systems are designed to act and can therefore assess the effect of interventions. As such, RL models can be designed to provide causal interpretations (Bareinboim et al. 2015, Zhang and Bareinboim 2017, Gasse et al. 2021).

[21] Given a causal graph, observational data can allow to identify average causal effects (rung 2), e.g. with the so-called backdoor criterion (Pearl 2009). For estimating counterfactuals (rung 3), assumptions beyond a causal graph and observational data must be met (Holland 1986, Peters et al. 2017).

6 Conclusion

Traditional scientific models were designed to satisfy elementwise representationality. This allowed scientists to directly inspect model elements to learn about Nature. Although ML models do not satisfy elementwise representationality, we have showed that it is still possible to learn about the phenomenon using them. All we need to do is to interrogate the model with suitable IML property descriptors. We have shown how such descriptors must be designed to enable scientific inference.

Funding

This work was supported by the Graduate School of Systemic Neuroscience (GSN) of the LMU Munich and by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A, and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC number 2064/1 – Project number 390727645.

Acknowledgements

We are very significantly indebted to Tom Sterkenburg, Seth Axen, Thomas Grote, Alexandra Gessner, Nacho Molina and Christian Scholbeck for their comments on the manuscript and their hints to related literature. Moreover, we thank the workshop participants of the Tübingen workshop "Philosophy of Science Meets Machine Learning" for the insightful comments and discussions after presenting some of the ideas described in the paper.

References

Aas, K., Jullum, M., and Løland, A. (2021). Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *Artificial Intelligence*, 298:103502.

Achinstein, P. (1974). Concepts of science. *Philosophy*, 49(187).

Alqaraawi, A., Schuessler, M., Weiß, P., Costanza, E., and Berthouze, N. (2020). Evaluating saliency map explanations for convolutional neural networks: a user study. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*, pages 275–285.

Anderson, C. (2008). The end of theory: The data deluge makes the scientific method obsolete. *Wired magazine*, 16(7):16–07.

Apley, D. W. and Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(4):1059–1086.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. (2019). Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115.

Bailer-Jones, D. M. (2003a). Models, theories and phenomena. *Proceedings of Logic Methodology and Philosophy of Science*.

Bailer-Jones, D. M. (2003b). When scientific models represent. *International studies in the philosophy of science*, 17(1):59–74.

Bailer-Jones, D. M. and Bailer-Jones, C. A. (2002). Modeling data: Analogies in neural networks, simulated annealing and genetic algorithms. In *Model-Based Reasoning*, pages 147–165. Springer.

Bareinboim, E., Forney, A., and Pearl, J. (2015). Bandits with unobserved confounders: A causal approach. *Advances in Neural Information Processing Systems*, 28.

Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. (2017). Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549.

Bau, D., Zhu, J.-Y., Strobelt, H., Zhou, B., Tenenbaum, J. B., Freeman, W. T., and Torralba, A. (2018). Gan dissection: Visualizing and understanding generative adversarial networks. *arXiv preprint arXiv:1811.10597*.

Bhatt, U., Xiang, A., Sharma, S., Weller, A., Taly, A., Jia, Y., Ghosh, J., Puri, R., Moura, J. M., and Eckersley, P. (2020). Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 648–657.

Bickler, S. H. (2021). Machine learning arrives in archaeology. *Advances in Archaeological Practice*, 9(2):186–191.

Boge, F. J. (2022). Two dimensions of opacity and the deep learning predicament. *Minds and Machines*, 32(1):43–75.

Bokulich, A. (2011). How scientific models can explain. *Synthese*, 180(1):33–45.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Breiman, L. et al. (2001). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231.

Buckner, C. (2018). Empiricism without magic: Transformational abstraction in deep convolutional neural networks. *Synthese*, 195(12):5339–5372.

Buckner, C. and Garson, J. (2019). Connectionism. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Fall 2019 edition.

Callender, C. (2015). One world, one beable. *Synthese*, 192(10):3153–3177.

Chen, H., Janizek, J. D., Lundberg, S., and Lee, S.-I. (2020). True to the model or true to the data? *arXiv preprint arXiv:2006.16234*.

Cichy, R. M. and Kaiser, D. (2019). Deep neural networks as scientific models. *Trends in cognitive sciences*, 23(4):305–317.

Constantinou, A. C., Liu, Y., Chobtham, K., Guo, Z., and Kitson, N. K. (2021). Large-scale empirical validation of bayesian network structure learning algorithms with noisy data. *International Journal of Approximate Reasoning*, 131:151–188.

Contessa, G. (2007). Scientific representation, interpretation, and surrogative reasoning. *Philosophy of science*, 74(1):48–68.

Cortez, P. and Silva, A. (2008). Using data mining to predict secondary school student performance. *EUROSIS*.

Covert, I., Lundberg, S. M., and Lee, S.-I. (2020). Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33:17212–17223.

Cox, D. R. (2006). *Principles of statistical inference*. Cambridge university press.

Cranmer, K., Brehmer, J., and Louppe, G. (2020). The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062.

Dandl, S., Molnar, C., Binder, M., and Bischl, B. (2020). Multi-objective counterfactual explanations. In *International Conference on Parallel Problem Solving from Nature*, pages 448–469. Springer.

Domingos, P. (2000). A unified bias-variance decomposition. In *Proceedings of 17th international conference on machine learning*, pages 231–238. Morgan Kaufmann Stanford.

Donnelly, J. and Roegiest, A. (2019). On interpretability and feature representations: an analysis of the sentiment neuron. In *European Conference on Information Retrieval*, pages 795–802. Springer.

Doshi-Velez, F. and Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Douglas, H. E. (2009). Reintroducing prediction to explanation. *Philosophy of Science*, 76(4):444–463.

Dwivedi, D., Nearing, G., Gupta, H., Sampson, A. K., Condon, L., Ruddell, B., Klotz, D., Ehret, U., Read, L., Kumar, P., et al. (2021). Knowledge-guided machine learning (kgml) platform to predict integrated water cycle and associated extremes. Technical report, Artificial Intelligence for Earth System Predictability.

Farrell, S., Calafiura, P., Mudigonda, M., Anderson, D., Vlimant, J.-R., Zheng, S., Bendavid, J., Spiropulu, M., Cerati, G., Gray, L., et al. (2018). Novel deep learning methods for track reconstruction. *arXiv preprint arXiv:1810.06111*.

Fisher, A., Rudin, C., and Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *J. Mach. Learn. Res.*, 20(177):1–81.

Friedman, J. H. et al. (1991). Multivariate adaptive regression splines. *The Annals of Statistics*, 19(1):1–67.

Frigg, R. and Nguyen, J. (2021). Scientific Representation. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Winter 2021 edition.

Gale, E. M., Martin, N., Blything, R., Nguyen, A., and Bowers, J. S. (2020). Are there any 'object detectors' in the hidden layers of cnns trained to identify objects or scenes? *Vision Research*, 176:60–71.

Gasse, M., Grasset, D., Gaudron, G., and Oudeyer, P.-Y. (2021). Causal reinforcement learning using observational and interventional data. *arXiv preprint arXiv:2106.14421*.

Gibson, P. B., Chapman, W. E., Altinok, A., Delle Monache, L., DeFlorio, M. J., and Waliser, D. E. (2021). Training machine learning models on climate model output yields skillful interpretable seasonal precipitation forecasts. *Communications Earth & Environment*, 2(1):1–13.

Gobet, F. (2018). Three views on expertise: Philosophical implications for rationality, knowledge, intuition and education. *Education and Expertise*, pages 58–74.

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *journal of Computational and Graphical Statistics*, 24(1):44–65.

Good, P. I. and Hardin, J. W. (2012). *Common errors in statistics (and how to avoid them)*. John Wiley & Sons.

Gühring, I., Raslan, M., and Kutyniok, G. (2020). Expressivity of deep neural networks. *arXiv preprint arXiv:2007.04759*.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Heinze-Deml, C., Maathuis, M. H., and Meinshausen, N. (2018). Causal structure learning. *Annual Review of Statistics and Its Application*, 5:371–391.

Heskes, T., Sijben, E., Bucur, I. G., and Claassen, T. (2020). Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models. *Advances in neural information processing systems*, 33:4778–4789.

Holland, P. W. (1986). Statistics and causal inference. *Journal of the American statistical Association*, 81(396):945–960.

Hooker, G. (2004). Discovering additive structure in black box functions. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 575–580.

Hooker, G. and Mentch, L. (2021). Bridging breiman's brook: From algorithmic modeling to statistical learning. *Observational Studies*, 7(1):107–125.

Hooker, G., Mentch, L., and Zhou, S. (2021). Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, 31(6):1–16.

Hutter, F., Kotthoff, L., and Vanschoren, J. (2019). *Automated machine learning: methods, systems, challenges*. Springer Nature.

Janzing, D., Minorics, L., and Blöbaum, P. (2020). Feature relevance quantification in explainable ai: A causal problem. In *International Conference on artificial intelligence and statistics*, pages 2907–2916. PMLR.

Jia, Y., Abbott, J. T., Austerweil, J. L., Griffiths, T., and Darrell, T. (2013). Visual concept learning: Combining machine vision and bayesian generalization on concept hierarchies. *Advances in Neural Information Processing Systems*, 26.

Kalisch, M. and Bühlmann, P. (2014). Causal structure learning and inference: a selective review. *Quality Technology & Quantitative Management*, 11(1):3–21.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al. (2018). Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR.

Kitchin, R. (2014). Big data, new epistemologies and paradigm shifts. *Big data & society*, 1(1):2053951714528481.

Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., and Liang, P. (2020). Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR.

König, G., Freiesleben, T., and Grosse-Wentrup, M. (2021a). A causal perspective on meaningful and robust algorithmic recourse. *arXiv preprint arXiv:2107.07853*.

König, G., Molnar, C., Bischl, B., and Grosse-Wentrup, M. (2021b). Relative feature importance. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9318–9325. IEEE.

Levy, A. (2012). Models, fictions, and realism: Two packages. *Philosophy of Science*, 79(5):738–748.

Lipton, Z. C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57.

Longino, H. E. (2018). *The fate of knowledge*. Princeton University Press.

Luan, H. and Tsai, C.-C. (2021). A review of using machine learning approaches for precision education. *Educational Technology & Society*, 24(1):250–266.

Luk, R. W. (2017). A theory of scientific study. *Foundations of Science*, 22(1):11–38.

Mayer-Schönberger, V. and Cukier, K. (2013). *Big data: A revolution that will transform how we live, work, and think*. Houghton Mifflin Harcourt.

McClelland, J. L., Rumelhart, D. E., Group, P. R., et al. (1987). *Parallel Distributed Processing, Volume 2: Explorations in the Microstructure of Cognition: Psychological and Biological Models*, volume 2. MIT press.

Mentch, L. and Hooker, G. (2016). Quantifying uncertainty in random forests via confidence intervals and hypothesis tests. *The Journal of Machine Learning Research*, 17(1):841–881.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

Molnar, C., Casalicchio, G., and Bischl, B. (2020a). Interpretable machine learning–a brief history, state-of-the-art and challenges. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 417–431. Springer.

Molnar, C., Freiesleben, T., König, G., Casalicchio, G., Wright, M. N., and Bischl, B. (2021). Relating the partial dependence plot and permutation feature importance to the data generating process. *arXiv preprint arXiv:2109.01433*.

Molnar, C., König, G., Bischl, B., and Casalicchio, G. (2020b). Model-agnostic feature importance and effects with dependent features–a conditional subgroup approach. *arXiv preprint arXiv:2006.04628*.

Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C. A., Casalicchio, G., Grosse-Wentrup, M., and Bischl, B. (2022). General pitfalls of model-agnostic interpretation methods for machine learning models. In Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.-R., and Samek, W., editors, *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*, pages 39–68, Cham. Springer International Publishing.

Mu, J. and Andreas, J. (2020). Compositional explanations of neurons. *Advances in Neural Information Processing Systems*, 33:17153–17163.

Nearing, G. S., Kratzert, F., Sampson, A. K., Pelissier, C. S., Klotz, D., Frame, J. M., Prieto, C., and Gupta, H. V. (2021). What role does hydrological science play in the age of machine learning? *Water Resources Research*, 57(3):e2020WR028091.

Nguyen, A., Yosinski, J., and Clune, J. (2016). Multifaceted feature visualization: Uncovering the different types of features learned by each neuron in deep neural networks. *arXiv preprint arXiv:1602.03616*.

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., and Carter, S. (2020). Zoom in: An introduction to circuits. *Distill*, 5(3):e00024–001.

Olah, C., Mordvintsev, A., and Schubert, L. (2017). Feature visualization. *Distill*, 2(11):e7.

Pearl, J. (2009). *Causality*. Cambridge university press.

Pearl, J. (2018). Theoretical impediments to machine learning with seven sparks from the causal revolution.

Pearl, J. and Mackenzie, D. (2018). *The book of why: the new science of cause and effect*. Basic books.

Peters, J., Bühlmann, P., and Meinshausen, N. (2016). Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(5):947–1012.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of causal inference: foundations and learning algorithms*. The MIT Press.

Pfister, N., Williams, E. G., Peters, J., Aebersold, R., and Bühlmann, P. (2021). Stabilizing variable selection and regression. *The Annals of Applied Statistics*, 15(3):1220–1246.

Raji, I. D., Smart, A., White, R. N., Mitchell, M., Gebru, T., Hutchinson, B., Smith-Loud, J., Theron, D., and Barnes, P. (2020). Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 33–44.

Räz, T. (2022). Understanding deep learning with statistical relevance. *Philosophy of Science*, 89(1):20–41.

Razavi, S. (2021). Deep learning, explained: Fundamentals, explainability, and bridgeability to process-based modelling. *Environmental Modelling & Software*, 144:105159.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM.

Ritchey, T. (2012). Outline for a morphology of modelling methods. *Acta Morphologica Generalis AMG Vol*, 1(1):1012.

Roscher, R., Bohn, B., Duarte, M. F., and Garcke, J. (2020). Explainable machine learning for scientific insights and discoveries. *IEEE Access*, 8:42200–42216.

Rosser, J. S. J. B. et al. (2008). A nobel prize for asymmetric information: the economic contributions of george akerlof, michael spence and joseph stiglitz. In *Leading Contemporary Economists*, pages 162–181. Routledge.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Salmon, W. C. (1979). Why ask,'why?'? an inquiry concerning scientific explanation. In *Hans Reichenbach: logical empiricist*, pages 403–425. Springer.

Salmon, W. C. (1998). *Causality and explanation*. Oxford University Press.

Schmidt, J., Marques, M. R., Botti, S., and Marques, M. A. (2019). Recent advances and applications of machine learning in solid-state materials science. *npj Computational Materials*, 5(1):1–36.

Scholbeck, C. A., Molnar, C., Heumann, C., Bischl, B., and Casalicchio, G. (2019). Sampling, intervention, prediction, aggregation: a generalized framework for model-agnostic interpretations. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 205–216. Springer.

Schwab, P. and Karlen, W. (2019). Cxplain: Causal explanations for model interpretation under uncertainty. *Advances in Neural Information Processing Systems*, 32.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. (2021). Towards causal representation learning.

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W., Bridgland, A., et al. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710.

Shahhosseini, M., Hu, G., and Archontoulis, S. V. (2020). Forecasting corn yield with machine learning ensembles. *arXiv preprint arXiv:2001.09055*.

Shmueli, G. et al. (2010). To explain or to predict? *Statistical science*, 25(3):289–310.

Slack, D., Hilgard, S., Jia, E., Singh, S., and Lakkaraju, H. (2020). Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 180–186.

Spinney, L. (2022). Are we witnessing the dawn of post-theory science? *The Guardian*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Stachl, C., Au, Q., Schoedel, R., Gosling, S. D., Harari, G. M., Buschek, D., Völkel, S. T., Schuwerk, T., Oldemeier, M., Ullmann, T., Hussmann, H., Bischl, B., and Bühner, M. (2020). Predicting personality from patterns of behavior collected with smartphones. *Proceedings of the National Academy of Sciences*, 117(30):17680–17687.

Stachowiak, H. (1973). *Allgemeine modelltheorie*. Springer.

Sterkenburg, T. F. and Grünwald, P. D. (2021). The no-free-lunch theorems of supervised learning. *Synthese*, 199(3):9979–10015.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., and Zeileis, A. (2008). Conditional variable importance for random forests. *BMC bioinformatics*, 9(1):1–11.

Štrumbelj, E. and Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665.

Sullivan, E. (2020). Understanding from machine learning models. *The British Journal for the Philosophy of Science*.

Suppes, P. (1966). Models of data. In *Studies in logic and the foundations of mathematics*, volume 44, pages 252–261. Elsevier.

Taleb, N. (2005). The black swan: Why don't we learn that we don't learn. *NY: Random House*.

Toulmin, S. E. (1961). *Foresight and understanding: An enquiry into the aims of science*. Greenwood Press.

Voss, C., Cammarata, N., Goh, G., Petrov, M., Schubert, L., Egan, B., Lim, S. K., and Olah, C. (2021). Visualizing weights. *Distill*, 6(2):e00024–007.

Wachter, S., Mittelstadt, B., and Floridi, L. (2017). Why a right to explanation of automated decision-making does not exist in the general data protection regulation. *International Data Privacy Law*, 7(2):76–99.

Wang, J., Wiens, J., and Lundberg, S. (2021). Shapley flow: A graph-based approach to interpreting model predictions. In *International Conference on Artificial Intelligence and Statistics*, pages 721–729. PMLR.

Watson, D. S. (2022). Conceptual challenges for interpretable machine learning. *Synthese*, 200(1):1–33.

Woodward, J. and Ross, L. (2021). Scientific Explanation. In Zalta, E. N., editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Summer 2021 edition.

Zaeem, M. N. and Komeili, M. (2021). Cause and effect: Concept-based explanation of neural networks. In *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2730–2736. IEEE.

Zednik, C. (2021). Solving the black box problem: a normative framework for explainable artificial intelligence. *Philosophy & Technology*, 34(2):265–288.

Zednik, C. and Boelsen, H. (2022). Scientific exploration and explainable artificial intelligence. *Minds and Machines*, pages 1–21.

Zhang, J. and Bareinboim, E. (2017). Transfer learning in multi-armed bandit: a causal approach. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pages 1778–1780.

Zhang, Z., Jin, Y., Chen, B., and Brown, P. (2019). California almond yield prediction at the orchard level with a machine learning approach. *Frontiers in Plant Science*, 10:809.

Zhao, Q. and Hastie, T. (2021). Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, 39(1):272–281.

Zhou, B., Sun, Y., Bau, D., and Torralba, A. (2018). Interpretable basis decomposition for visual explanation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 119–134.

2.3 Paper III: Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process

Molnar*, Christoph, Timo Freiesleben*, Gunnar König*, Julia Herbinger, Tim Reisinger, Giuseppe Casalicchio, Marvin Wright, and Bernd Bischl. **Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process.** *The 1st World Conference on eXplainable Artificial Intelligence (2023).*

Gunnar König contributed to the paper as shared first author. Gunnar König and Timo Freiesleben led the revision of the paper and made conceptual contributions, particularly in Section 2.5, and contributed some of the proofs, particularly Theorem 3. Gunnar König implemented the application example and wrote sections 1.1 and 4. Christoph Molnar developed the initial idea and wrote large parts of the paper. Christoph Molnar and Marvin Wright implemented and ran the simulation study in Section 3.1; Julia Herbinger and Tim Reisinger implemented and ran the comparison with model-based approaches in Section 3.2. All authors added valuable new discussion points and helped revise the text.

Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process

Christoph Molnar*[1,4][0000-0003-2331-868X], Timo Freiesleben*[2][0000-0003-1338-3293], Gunnar König*[1,3][0000-0001-6141-4942], Julia Herbinger[1,7], Tim Reisinger[1], Giuseppe Casalicchio[1,7][0000-0001-5324-5966], Marvin N. Wright[4,5,6][0000-0002-8542-6291], and Bernd Bischl[1,7][0000-0001-6002-6980]

[1] Department of Statistics, LMU Munich, Munich, Germany
[2] Cluster of Excellence Machine Learning, Tübingen, Germany
[3] University of Vienna, Vienna, Austria
[4] Leibniz Institute for Prevention Research and Epidemiology, Bremen, Germany
[5] University of Bremen, Bremen, Germany
[6] University of Copenhagen, Copenhagen, Denmark
[7] Munich Center for Machine Learning (MCML)

Abstract Scientists and practitioners increasingly rely on machine learning to model data and draw conclusions. Compared to statistical modeling approaches, machine learning makes fewer explicit assumptions about data structures, such as linearity. Consequently, the parameters of machine learning models usually cannot be easily related to the data generating process. To learn about the modeled relationships, partial dependence (PD) plots and permutation feature importance (PFI) are often used as interpretation methods. However, PD and PFI lack a theory that relates them to the data generating process. We formalize PD and PFI as statistical estimators of ground truth estimands rooted in the data generating process. We show that PD and PFI estimates deviate from this ground truth not only due to statistical biases, but also due to learner variance and Monte Carlo approximation errors. To account for these uncertainties in PD and PFI estimation, we propose the learner-PD and the learner-PFI based on model refits and propose corrected variance and confidence interval estimators.

Keywords: XAI, Interpretable Machine Learning, Permutation Feature Importance, Partial Dependence Plot, Statistical Inference, Uncertainty Quantification

1 Introduction

Statistical models such as linear or logistic regression models are frequently used to learn about relationships in data. Assuming that a statistical model reflects

* equal contribution

the data generating process (DGP) well, we may interpret the model coefficients in place of the DGP and draw conclusions about the data. An important part of interpreting the coefficients is the quantification of their uncertainty via standard errors, which allows separation of random noise (non-significant coefficients) from real effects.

Increasingly, machine learning (ML) approaches – such as gradient-boosted trees, random forests or neural networks – are being used in science instead of or in addition to statistical models as they are able to learn highly-non linear relationships and interactions automatically. Applications range from modeling volunteer labor supply [4], mapping fish biomass [17], analyzing urban reservoirs [36], identifying disease-associated genetic variants [8], to inferring behavior from smartphone use [43]. However, in contrast to statistical models, machine learning approaches often lack a mapping between model parameters and properties of the DGP. This is problematic, since in scientific applications the model is only the means to an end: a better understanding of the DGP, in particular to learn what features are predictive of the target variable.

Interpretation methods [41] are a (partial) remedy to the lack of interpretable parameters of more complex models. Model-agnostic techniques, such as partial dependence (PD) plots [20] and permutation feature importance (PFI) [9,18] can be applied to any ML model and are popular methods for describing the relationship between input features and model outcome on a global level. PD plots visualize the average effect that features have on the prediction, and PFI estimates how much each feature contributes to the model performance and therefore how relevant a feature is.

Scientists who want to use PD and PFI to draw conclusions about the DGP face a problem as these methods have been designed to describe the prediction function, but lack a theory linking them to the DGP. In particular, the uncertainty of PD and PFI with respect to the DGP is not quantified, making it hard for scientists to assess the extent to which it is justified to draw conclusions based on the PD and PFI.

Contributions We are the first to treat PD and PFI as statistical estimators of ground truth properties in the DGP. We introduce two notions, model-PD/PFI and learner-PD/PFI, which allow to analyze the uncertainty due to Monte-Carlo integration and uncertainty due to the training data/process, respectively. We perform bias-variance decompositions and propose theorems of unbiasedness, standard estimators, and confidence intervals for both PD and PFI. In addition, we leverage a variance correction approach from model performance estimation [35] to adjust for variance underestimation due to sample dependency.

Structure We start with a motivating example (Section 1.1) and a discussion of related work (Section 1.2). In the methods section (Section 2), we introduce PD and PFI formally, relate them to the DGP, and provide bias-variance decompositions, variance estimators and confidence intervals. In the simulation study in Section 3, we test our proposed methods in various settings and compare them to alternative approaches. In the application in Section 4, we revisit the motivating example to demonstrate how our confidence intervals for PD/PFI may help scientists to draw more justified conclusions about the DGP. Finally, we discuss the limitations of our work in Section 5.

1.1 Motivating Example

Imagine a researcher who wants to use machine learning methods and the publicly available UCI heart disease dataset [15] ($n = 918$) not only to predict heart disease, but also to understand how the disease is associated with sociological and medical indicators.

To select the model class, she compares the performance w.r.t. the predicted probabilities of a logistic regression model, a decision tree (CART) [10], and a random forest classifier [9] using 5-fold cross validation measured by the Brier score on the dataset; the mean losses for the different models are 0.130 (logistic regression), 0.258 (tree), and 0.125 (random forest). Since the random forest outperforms the linear model and decision tree, she uses a random forest for further analysis; she fits the model on 60 per cent of the data and uses the remaining 40 per cent as test set.[8]

To learn about the associations in the data, she applies the PD and PFI. To get interpretations that are true to the data and that avoid extrapolation, she employs conditional sampling based versions of PD and PFI (for a discussion of marginal versus conditional sampling, we refer to the literature [13,19], Section 2.1, and Section 2.3). The conditional PD corresponds to the expected prediction and therefore indicates how the probability of having heart disease varies with the feature of interest [19]. Conditional feature importance quantifies the surplus contribution of each feature over the remaining features (and can be linked to conditional dependence with the prediction target [28,45]).[9]

The results (Figure 1) match the researcher's intuition. Many conditional PFI values are small, indicating that the features could be replaced with the remaining features. The most important features are the slope of the ECG segment (`STSlope`), the type of chest pain (`ChestPainType`), and cholesterol level (`Cholesterol`). Furthermore, the researcher is interested in the relationship between heart disease and age. Thus, she inspects the corresponding conditional PD plot. She observes that the probability of having chronic heart disease increases with age and that there is a small bump around the age of 55.

Although the researcher finds the results plausible, she is unsure whether her conclusions extend to the data generating process (DGP). Are features

[8] All code is publicly available as part of the supplementary material.

[9] Conditional interpretation methods require sampling from conditional distributions. She samples categorical variables using a log-loss optimal classifier, and samples continuous variables by predicting the conditional mean and resampling residuals (thereby assuming homoscedasticity). She fits a random forest once on the dataset for all sampling tasks. To model multivariate mixed distributions, she employs a sequential design [5,7].

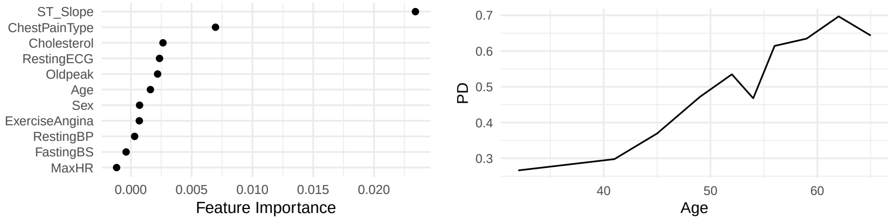


Figure 1: Left: Conditional Feature Importance. Right: Conditional Partial Dependence Plot for the feature `Age`. The values are difficult to interpret since it is unclear how uncertainties in model fitting and IML method estimation influence them.

with nonzero feature importance actually relevant, or are the values nonzero by chance? Does the shape of the PDP really reflect the data? After all, various uncertainties could influence her result: The feature importance and conditional PD results vary when they are recomputed — even for the same model; and the random forest fit itself is a random variable as well.

Throughout this paper, we propose confidence intervals for partial dependence and feature importance values that take the uncertainties from the estimation of the interpretability method and the model fitting into account. We will return to this example in Section 4 and Figure 6, where we show how our approach can help the researcher to evaluate the uncertainty in her estimates.

1.2 Related Work

PD: For models with inherent variance estimators (such as Bayesian additive regression trees) it is possible to construct model-based confidence intervals [11]. Moosbauer et al. [34] introduced a variance estimator for PD which is applicable to all probabilistic models that provide information on posterior (co)variance, such as Gaussian Processes (GPs). Furthermore, various applied articles contain computations of PD confidence bands [4,22,17,16,37,36]. These approaches either quantify only the error due to Monte Carlo approximation or do not account for underestimation of the variance when covering learner variance. This demonstrates the need for a theoretical underpinning of this inferential tool for practical research.

PFI: Various proposals for confidence intervals and variance estimation exist. Many of them are specific to the random forest PFI [26,3,27], for which Altmann et al. [1] propose a test for null importance. There are also model-agnostic accounts that are more similar to our work [45,46,47], however, unlike these other proposals, we additionally correct for variance underestimation arising from resampling [35] and relate the estimators to the proposed ground truth PFI. An alternative approach for providing bounds on PFI is proposed by Fisher et al. [18] via Rashomon sets, which are sets of models with similar near-optimal prediction accuracy. Our approach differs since our bounds are relative to a fixed model or learning process, whereas Rashomon sets are defined exclusively by the model performance. Furthermore, alternative approaches of "model-free" inference have been introduced [39,38,48], which aim to infer properties of the data without an intermediary machine learning model.

2 Methods

In this section, we present our formal framework: We introduce notation and background on PD and PFI (Section 2.1); formulate PD and PFI as estimators of (proposed) ground truth estimands in the DGP (Section 2.3); apply bias and variance decompositions and separate different sources of uncertainty (Section 2.4); and propose variance estimators and confidence intervals for the model-PD/PFI (which only takes the variance from Monte-Carlo integration into account, see Section 2.5) and the learner-PD/PFI (which also takes learner variance into account, see Section 2.6).

2.1 Notation

We denote the joint distribution induced by the data generating process as $\mathbb{P}_{XY}$, where X is a p-dimensional random variable and Y a 1-dimensional random variable. We consider the case where we aim to describe the true mapping from X to the target Y with $f(X) = E[Y \mid X = x]$.[10] We denote a single random draw from the DGP with $x^{(i)}$ and $y^{(i)}$, and a dataset consisting of n draws $\mathcal{D}_n$.

A machine learning model $\hat{f}$ is a function ($\hat{f} : \mathcal{X} \to \mathcal{Y}$) that maps a vector x from the feature space $\mathcal{X} \subseteq \mathbb{R}^p$ to a prediction $\hat{y}$ (e.g. in $\mathcal{Y} = \mathbb{R}$ for regression). The model $\hat{f}$ is induced based on a dataset $\mathcal{D}_n$, using a loss function $L : \mathcal{Y} \times \mathbb{R}^p \to \mathbb{R}_0^+$. The model $\hat{f}$ is induced by the learner algorithm $I : \Delta \to \mathcal{H}$ that maps from the space of datasets Δ to the function hypothesis space $\mathcal{H}$. The learning process contains an essential source of randomness, namely the training data. Since the model $\hat{f}$ is induced by the learner fed with data, it can be seen as a realization of a random variable F with distribution $\mathbb{P}_F$. We assume that the model is evaluated with a risk function $\mathcal{R}(\hat{f}) = \mathbb{E}_{XY}[L(Y, \hat{f}(X))] = \int L(y, \hat{f}(x)) d\mathbb{P}_{XY}$. The dataset $\mathcal{D}_n$ is split into $\mathcal{D}_{n_1}$ for model training and $\mathcal{D}_{n_2}$ for evaluation. The empirical risk is estimated with $\hat{\mathcal{R}}(\hat{f}_{\mathcal{D}_{n_2},\lambda}) := \frac{1}{n_2} \sum_{i=1}^{n_2} L\left(y^{(i)}, \hat{f}_{\mathcal{D}_{n_2},\lambda}(x^{(i)})\right)$.

Many interpretation techniques require perturbing variables by resampling from marginal or conditional distributions. We use ϕ to denote a sampler, which can formally be seen as a density function. A dataset drawn with a marginal sampler (denoted ϕ_{marg}) follows $P(X_j)$, and a dataset drawn with a conditional sampler (denoted ϕ_{cond}) follows $P(X_j|X_C)$. The choice of the sampler affects the interpretation of PD and PFI [33,32,18,45,2] and should depend on the modeler's objective. Under certain conditions, the marginal sampler allows to estimate

[10] This choice for f is motivated by the fact that the conditional expectation is the Bayes-optimal predictor for the L2 loss and for the log-loss optimal predictor in binary classification [24].

causal effects [49], but for correlated input features, the marginal sampler may create unrealistic data and the conditional sampler may be a better choice to draw inference [19] (see online Appendix A [31] for details).

2.2 Interpretation Techniques

Partial Dependence Plot The PD of a feature set X_S, $S \subseteq \{1, \ldots, p\}$ (usually $\mid S \mid = 1$) for a given $x \in X_S$, a model $\hat{f}$ and a sampler $\phi : \mathcal{X}_S \rightarrow \{\psi \mid \psi$ density on $\mathcal{X}_C\}$ is:

$$PD_{S,\hat{f},\phi}(x) := \mathbb{E}_{\tilde{X}_C \sim \phi(x)}[\hat{f}(x, \tilde{X}_C)] = \int_{\tilde{x}_c \in \tilde{\mathcal{X}}_C} \phi(x)(\tilde{x}_c)\hat{f}(x, \tilde{x}_c)\, d\tilde{x}_c, \qquad (1)$$

where $\tilde{X}_C$ is a random variable distributed with density $\phi(x)$, and C denote the indices of the remaining features so that $S \cup C = \{1, \ldots, p\}$ and $S \cap C = \emptyset$.
To estimate the PD for a specific function $\hat{f}$ using Monte Carlo integration, we draw $r \in \mathbb{N}$ samples for every $x \in \mathcal{X}_S$ from $\phi(x)$ and denote the corresponding dataset by $B_{\phi(x)} = (\tilde{x}_C^{(i,x)})_{i=1,\ldots,r}$. The estimation is given by:

$$\widehat{PD}_{S,\hat{f},\phi}(x) = \frac{1}{r}\sum_{i=1}^{r} \hat{f}(x, \tilde{x}_C^{(i,x)}). \qquad (2)$$

By partial dependence plot (PDP) we denote the graph that visualizes the PDP. The PDP consists of a line connecting the points $\{(x^{(g)}, \widehat{PD}_{S,\hat{f},\phi}(x^{(g)})\}_{g=1}^{G}$, with G grid points that are usually equidistant or quantiles of $\mathbb{P}_{X_S}$. See Figure 1 for an example of a PDP.
For the marginal sampler, the PDP of a model $\hat{f}$ visualizes the expected effect of a feature after marginalizing out the effects of all other features [20]. For the conditional sampler, the PDP is also called M-plot and visualizes the expected prediction given the features of interest, taking into account its associative dependencies with all other features [20,2].

Permutation Feature Importance The PFI of a feature set X_S (usually just one feature) for a model $\hat{f}$ and a sampler $\phi : \mathcal{X}_C \rightarrow \{\psi \mid \psi$ density on $\mathcal{X}_S\}$ is defined by:

$$PFI_{S,\hat{f},\phi} := \mathbb{E}_{X_C,Y}[\mathbb{E}_{\tilde{X}_S \sim \phi(X_C)}[L(Y, \hat{f}(\tilde{X}_S, X_C))]] - \mathbb{E}_{XY}[L(Y, \hat{f}(X))], \qquad (3)$$

where $\tilde{X}_S$ is a random variable distributed with density $\phi(X_C) \sim P(X_S|X_C)$, and X_C are the remaining features $\{1, \ldots, p\} \setminus S$. To estimate the PFI for a specific function $\hat{f}$ and a sampler ϕ using Monte Carlo integration, we draw $r \in \mathbb{N}$ samples for every datapoint $x_C^{(i)} \in \mathcal{X}_C$ ($x_C^{(i)}$ describes the feature values in C of the i-th instance in the evaluation[11] dataset D_{n_2}) from $\phi(x_C^{(i)})$ and denote

[11] The estimation of $\widehat{PFI}$ requires unseen data, so that the loss estimates deliver unbiased results [29,14].

the corresponding datasets by $B_{\phi(x_C^{(i)})} = (\tilde{x}_S^{(k,i)})_{k=1,\ldots,r}$. The estimation is given by:

$$\widehat{PFI}_{S,\hat{f},\phi} = \frac{1}{n_2}\sum_{i=1}^{n_2}\left(\frac{1}{r}\sum_{k=1}^{r} L(y^{(i)}, \hat{f}(\tilde{x}_S^{(k,i)}, x_C^{(i)})) - L(y^{(i)}, \hat{f}(x^{(i)}))\right). \quad (4)$$

We restrict PFI to losses that can be computed per instance.[12] See Figure 1 for a PFI example.

If we resample the perturbed variables from the marginal distribution, the PFI of a model $\hat{f}$ describes the change in loss if the feature values in X_S are randomly sampled from X_S i.e. the possible dependence to X_C and Y is broken (extrapolation) [9,18]. If we sample X_S conditional on the remaining variables X_C, PFI is also called the conditional PFI and may be interpreted as the *additional* importance of a feature *given that we already know the other feature values* [32,45,25,12].

Indices To avoid indices overhead and because PDP/PFI and their respective estimations are always relative to a fixed feature set S and sampler ϕ, we will abbreviate $PD_{S,\hat{f},\phi}, \widehat{PD}_{S,\hat{f},\phi}, PFI_{S,\hat{f},\phi}, \widehat{PFI}_{S,\hat{f},\phi}$ with $PD_{\hat{f}}, \widehat{PD}_{\hat{f}}, PFI_{\hat{f}}, \widehat{PFI}_{\hat{f}}$ respectively.

2.3 Relating the Model to the Data Generating Process

The goal of statistical inference is to gain knowledge about DGP properties via investigating model properties. For example, under certain assumptions, the coefficients of a generalized linear model (i.e. model properties) can be related to parameters of the respective conditional distribution defined by the DGP, such as conditional mean and covariance structure (i.e. DGP properties). Unfortunately, machine learning models such as random forests or neural networks lack such a mapping between learned model parameters and DGP properties. Interpretation methods such as PD and PFI provide **external descriptors** of how features affect the model predictions. However, PD and PFI are estimators that lack a counterpart estimand in the DGP.

We define the ground truth version of PD and PFI, we call them *DGP-PD* and the *DGP-PFI*, as the PD and PFI applied to the true function f instead of the trained model $\hat{f}$:

Definition 1 (DGP-PD). *The DGP-PD is the PD applied to function* $f : \mathcal{X} \mapsto \mathcal{Y}$ *of the DGP with sampler* $\phi : \mathcal{X}_S \to \{\psi \mid \psi \text{ density on } \mathcal{X}_C\}$.

$$\textit{DGP-PD}(x) := PD_f(x)$$

Definition 2 (DGP-PFI). *The DGP-PFI is the PFI applied to function* $f : \mathcal{X} \mapsto \mathcal{Y}$ *of the DGP with sampler* $\phi : \mathcal{X}_C \to \{\psi \mid \psi \text{ density on } \mathcal{X}_S\}$.

$$\textit{DGP-PFI} := PFI_f$$

[12] This excludes losses such as the area under the receiver operating characteristic curve (AUC).

Note that the DGP-PD and DGP-PFI may not be well-defined for all possible samplers. The DGP $f(x) = \mathbb{E}[Y \mid X = x]$ for instance is undefined for $x \in \mathcal{X}$ with zero density ($\psi_X(x) = 0$). For the marginal sampler, for instance, DGP-PD and DGP-PFI might not be defined if the input features show strong correlations [25]. Conditional samplers, on the other side, do not face this threat as they preserve dependencies between features and therefore do not create unrealistic inputs [32,18,45,2].[13] However, under certain conditions, it can still be useful to also use other samplers than the conditional samplers to gain insight into the DGP. For example, under certain conditions, the marginal PDP allows to estimate causal effects [49] or recover relevant properties of linear DGPs [23].

Clearly, the function f is unknown in most applications, which makes it impossible to know the DGP-PD and DGP-PFI for these cases. However, Definitions 1 and 2 enable, at least in theory, to compare the PD/PFI of a model with the PD/PFI of the DGP **in simulation studies** and to research statistical biases. More importantly, the ground truth definitions of DGP-PD and DGP-PFI allow us to treat PD and PFI as statistical estimators of properties of the DGP.

In this work, we study PD and PFI as statistical estimators of the ground truth DPG-PD and DGP-PFI – including bias and variance decompositions – as well as confidence interval estimators. DGP-PD and DGP-PFI describe interesting properties of the DGP concerning the associational dependencies between the predictors and the target [19]; however, practitioners must decide whether these properties are relevant to answer their question or if different tools of model-analysis provide more interesting estimands.

2.4 Bias-Variance Decomposition

The definition of DGP-PD and DGP-PFI gives us a ground truth to which the PD and PFI of a model can be compared – at least in theory and simulation. The error of the estimation (mean squared error between estimator and estimand) can be decomposed into the systematic deviation from the true estimand (statistical bias) and the learner variance. PD and PFI are both expectations over the (usually unknown) joint distribution of the data. The expectations are therefore typically estimated from data using Monte Carlo integration, which adds another source of variation to the PFI and PD estimates. Figure 2 visualizes the chain of errors that stand between the estimand (DGP-PD, DGP-PFI) and the estimates ($\widehat{PD}$, $\widehat{PFI}$).

For the PD, we compare the mean squared error (MSE) between the true DGP-PD (PD_f as defined in Equation 1) with the theoretical PD of a model

[13] To illustrate the idea of unrealistic data points, think of two strongly correlated features such as the weight and height of a person. Not every combination of feature values is possible – a person with a weight of 4kg and a height of 2m is from a biological perspective inconceivable.

$$\begin{tikzcd}
\text{DGP-PD} \arrow[r, "\text{Bias}"] & \text{learner-PD} \arrow[r, "\text{Variance}"] & \text{model-PD} \arrow[r, "\text{MC}"] & \widehat{PD} \\
f \arrow[u] \arrow[d] \arrow[r, "\text{Learner Bias}"] & \mathbb{E}[\hat{f}] \arrow[u] \arrow[d] \arrow[r, "\text{Learner Variance}"] & \hat{f} \arrow[u] \arrow[d] & \\
\text{DGP-PFI} \arrow[r, "\text{Bias}"] & \text{learner-PFI} \arrow[r, "\text{Variance}"] & \text{model-PFI} \arrow[r, "\text{MC}"] & \widehat{PFI}
\end{tikzcd}$$

Figure 2: A model $\hat{f}$ deviates from f due to learner bias and variance. Similarly, $\widehat{PD}$ and $\widehat{PFI}$ estimates deviate from their ground truth versions DGP-PD and DGP-PFI due to bias, variance, and Monte Carlo integration (MC).

instance $\hat{f}$ ($PD_{\hat{f}}$) at position x.

$$\mathbb{E}_F[(PD_f(x) - PD_{\hat{f}}(x))^2] = \underbrace{(PD_f(x) - \mathbb{E}_F[PD_{\hat{f}}(x)])^2}_{Bias^2} + \underbrace{\mathbb{V}_F[PD_{\hat{f}}(x)]}_{Variance}$$

Here, F is the distribution of the trained models, which can be treated as a random variable. The bias-variance decomposition of the MSE of estimators is a well-known result [21]. For completeness, we provide a proof in online Appendix B [31]. Figure 3 visualizes bias and variance of a PD curve, and the variance due to Monte Carlo integration.

Similarly, the MSE of the theoretical PFI of a model (Equation 3) can be decomposed into squared bias and variance. The proof can be found in online Appendix C [31].

$$\mathbb{E}_F[(PFI_{\hat{f}} - PFI_f)^2] = Bias^2_F[PFI_{\hat{f}}] + \mathbb{V}_F[PFI_{\hat{f}}]$$

The learner variance of PD/PFI stems from variance in the model fit, which depends on the training sample. When constructing confidence intervals, we must take into account the variance of PFI and PDP across model fits, and not just the error due to Monte Carlo integration. As we show in an application (Section 4), whether PD and PFI are based on a single model or are averaged across model refits can impact both the interpretation and especially the certainty of the interpretation. We therefore distinguish between model-PD/PFI and learner-PD/PFI, which are averaged over refitted models. Variance estimators for model-PD/PFI only account for variance due to Monte Carlo integration.

2.5 Model-PD and Model-PFI

Here, we study the model-PD and the model-PFI, and provide variance and confidence interval estimators. With the terms model-PD and model-PFI, we refer to the original proposals for PD [20] and PFI [9,18] for fixed models. Conditioning on a given model $\hat{f}$ ignores the learner variance due to the learning process. Only the variance due to Monte Carlo integration can be considered in this case.

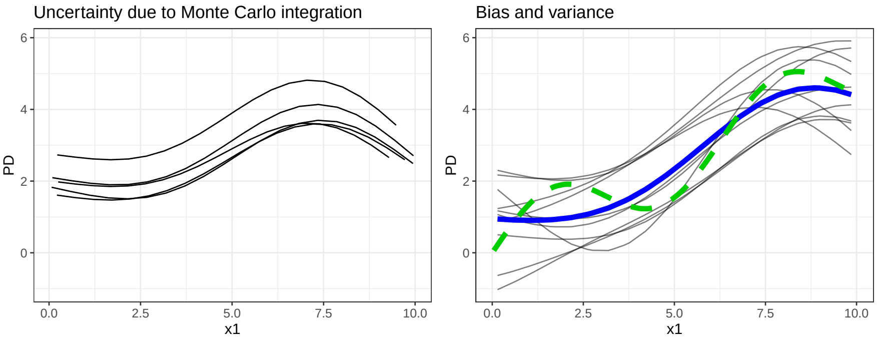


Figure 3: Illustration of bias, variance and Monte Carlo approximation for the PD with marginal sampling. Left: Various PDPs using different data for the Monte Carlo integration, but keeping the model fixed. Right: The green dashed line shows the DGP-PDP of a toy example. Each thin line is the PDP for the model fitted with a different sample, and the thick blue line is the average thereof. Deviations of the DGP-PDP from the expected PDP are due to bias. Deviations of the individual model-PDPs from the expected PDP are due to learner variance.

The model-PD estimator (Equation (2)) is unbiased regarding the theoretical model-PD (Equation (1)). Similarly, the estimated model-PFI (Equation 4) is unbiased with respect to the theoretical model-PFI (Equation 3). These findings rely on general properties of Monte Carlo integration, which state that Monte Carlo integration converges to the integral due to the law of large numbers. Proofs can be found in online Appendix D and F [31]. Moreover, under certain conditions, model-PD and model-PFI are unbiased estimators of the DGP-PD (Theorem 1) and DGP-PFI (Theorem 2), respectively.

To quantify the variance due to Monte Carlo integration and to construct confidence intervals, we calculate the variance across the sample. For the model-PD, the variance can be estimated with:

$$\widehat{\mathbb{V}}(\widehat{PD}_{\hat{f}}(x)) = \frac{1}{r(r-1)} \sum_{i=1}^{r} \left(\hat{f}(x, \tilde{x}_C^{(i,x)}) - \widehat{PD}_{\hat{f}}(x) \right)^2 . \tag{5}$$

Similarly for the model-PFI, the variance can be estimated with:

$$\widehat{\mathbb{V}}(\widehat{PFI}_{\hat{f}}) = \frac{1}{n_2(n_2-1)} \sum_{i=1}^{n_2} \left(L^{(i)} - \widehat{PFI}_{\hat{f}} \right)^2 , \tag{6}$$

where $L^{(i)} = \frac{1}{r} \sum_{k=1}^{r} L(y^{(i)}, \hat{f}(\tilde{x}_S^{(k,i)}, x_C^{(i)})) - L(y^{(i)}, \hat{f}(x^{(i)}))$.

The model-PD and model-PFI are mean estimates of independent samples with estimated variance. As such, they can be modelled approximately with a t-distribution with $r-1$ and n_2-1 degrees of freedom, respectively. This allows us to construct point-wise confidence bands for the model-PD and confidence intervals for the model-PFI that capture the Monte Carlo integration uncertainty. We define point-wise $1-\alpha$-confidence bands around the estimated model-PD:

$$CI_{\widehat{PD}_{\hat{f}}(x)} = \left[\widehat{PD}_{\hat{f}}(x) \pm t_{1-\frac{\alpha}{2}} \sqrt{\widehat{\mathbb{V}}(\widehat{PD}_{\hat{f}}(x))}\right]. \tag{7}$$

where $t_{1-\frac{\alpha}{2}}$ is the $1-\alpha/2$ quantile of the t-distribution with $r-1$ degrees of freedom. We proceed in the same manner for PFI but with $n_2 - 1$ degrees of freedom:

$$CI_{\widehat{PFI}_{\hat{f}}} = \left[\widehat{PFI}_{\hat{f}} \pm t_{1-\frac{\alpha}{2}} \sqrt{\widehat{\mathbb{V}}(\widehat{PFI}_{\hat{f}})}\right]. \tag{8}$$

Confidence intervals for model-PD and model-PFI ignore the learner variance. Therefore, the interpretation is limited to variance regarding the Monte Carlo integration, and we cannot generalize results to the DGP. The model-PD/PFI and their confidence bands/intervals are applicable when the focus is a fixed model.

2.6 Learner-PD and Learner-PFI

To account for the learner variance, we propose the learner-PD and the learner-PFI, which average the PD/PFI over m model fits $\hat{f}_d$ with $d \in \{1, \dots, m\}$. The models are produced by the same learning algorithm, but trained on different data samples, denoted by training sample indices B_d and the remaining test data B_{-d} so that $B_d \cap B_{-d} = \emptyset$ and $B_d \cup B_{-d} = \mathcal{D}_n$. The learner-variants are averages of the model-variants, where for each model-PD/PFI, the model is repeatedly "sampled" from the distribution of models F.

The learner-PD is therefore the expected PD over the distribution of models generated by the learning process, i.e. $\mathbb{E}_F[PD_{\hat{f}}(x)]$. We estimate the learner-PD with:

$$\overline{\widehat{PD}}(x) = \frac{1}{m} \sum_{d=1}^{m} \frac{1}{r} \sum_{i=1}^{r} \hat{f}_d \left(x, x_C^{i,x,d}\right), \tag{9}$$

where $\hat{f}_d$ is trained on sample indices B_d and the PD estimated with data $B_{\phi(x),d}$ using a sampler ϕ m-times.

Following the PD, the learner-PFI is the expected PFI over the distribution of models produced by the learner: $\mathbb{E}_F[PFI_{\hat{f},\phi}]$. We propose the following estimator for the learner-PFI:

$$\overline{\widehat{PFI}} = \frac{1}{m} \sum_{d=1}^{m} \frac{1}{n_2} \sum_{i=1}^{n_2} \left(\bar{\tilde{L}}_d^{(i)} - L_d^{(i)}\right), \tag{10}$$

where losses $L_d^{(i)} = L(y^{(i)}, \hat{f}_d(x^{(i)}))$ and $\bar{\tilde{L}}_d^{(i)} = \frac{1}{r}\sum_{k=1}^{r} L(y^{(i)}, \hat{f}_d(\tilde{x}_S^{(k,i,d)}, x_C^{(i)}))$ are estimated with data B_{-d} and m-times sampled data $B_{\phi(x),d}$ for a model trained on data B_d. A similar estimator has been proposed by Janitza et al. [27] for random forests.

Bias of the Learner-PD The learner-PD is an unbiased estimator of the expected PD over the distribution of models F, since

$$\mathbb{E}_F[\overline{\widehat{PD}}(x)] = \mathbb{E}_F\left[\frac{1}{m}\sum_{d=1}^{m}\widehat{PD}_{\hat{f}_d}(x)\right] = \frac{m}{m}\mathbb{E}_F[PD_{\hat{f}_d}(x)] = \mathbb{E}_F[PD_{\hat{f}_d}(x)].$$

The bias of the learner-PD *regarding the DGP-PD* is linked to the bias of the learner. If the learner is unbiased, the PDs are unbiased as well.

Theorem 1. *Learner unbiasedness implies PD unbiasedness:*
$\mathbb{E}_F[\hat{f}(x)] = f(x) \implies \mathbb{E}_F[PD_{\hat{f}}(x)] = PD_f(x)$

Proof Sketch 1. *Applying Fubini's Theorem allows us to switch the order of integrals. Further replacing $\mathbb{E}_F[\hat{f}(x)]$ with f proves the unbiasedness. A full proof can be found in online Appendix E [31].*

By learner bias, we refer to the expected deviation between the estimated $\hat{f}$ and the true function f. Particularly interesting in this context is the inductive bias (i.e. the preference of one generalization over another) that is needed for learning ML models that generalize [30]. A wrong choice of inductive bias, such as searching models $\hat{f}$ in a linear hypotheses class when f is non-linear, leads to deviations of the expected $\hat{f}$ from f. But there are also other reasons why a bias of $\hat{f}$ from f may occur, for example if using an insufficiently large sample of training data. We discuss the critical assumption of learner unbiasedness further in Section 5.

Bias of the Learner-PFI The learner-PFI is unbiased regarding the expected learner-PFI over the distribution of models F, since the learner-PFI is a simple mean estimate. However, unlike the learner-PD, learner unbiasedness does not generally imply unbiasedness of the learner-PFI *regarding the DGP-PFI.* This is generally only the case, if we use the conditional sampler.

Theorem 2. *If the learner is unbiased with $\mathbb{E}_F[\hat{f}] = f$ and the L2-loss is used, then the conditional model-PFI and conditional learner-PFI are unbiased estimators of the conditional DGP-PFI.*

Proof Sketch 2. *Both L and $\tilde{L}$ can be decomposed into bias, variance, and irreducible error. Due to the subtraction, the irreducible error vanishes, and the differences of biases and variances remain. Model unbiasedness sets the bias terms to zero and variance becomes zero due to conditional sampling. The extended proof can be found in online Appendix G [31].*

Intuitively, the model-PFI and learner-PFI should tend to have a negative bias and therefore underestimate the DGP-PFI. A model cannot use more information about the target than what is encoded in the DGP. However, as Theorem 3 shows, under specific conditions, the PFI using conditional sampling can be larger than the DGP-PFI.

Theorem 3. *The difference between the conditional model-PFI and the conditional DGP-PFI is given by:*

$$PFI_f - PFI_{\hat{f}} = 2\mathbb{E}_{X_C}\left[\mathbb{V}_{X_S|X_C}[f] - Cov_{X_S|X_C}[f, \hat{f}]\right].$$

Proof Sketch 3. *For the L2 loss, the expected loss of a model $\hat{f}$ can be decomposed into the expected loss between $\hat{f}$ and f and the expected variance of Y given X. Due to the subtraction, the latter term vanishes. The remainder can be simplified using that $Y \perp\!\!\!\perp \tilde{X}_S \mid X_C$ and $P(\tilde{X}_S, X_C) = P(X_S, X_C)$ due to the conditonal sampling. The extended proof can be found in online Appendix H [31].*

However, for an overestimation of the conditional PFI to occur, the expected conditional variance of $\hat{f}$ must be greater than the one of f. Moreover, $\hat{f}$ and f must have a large expected conditional covariance, meaning that $\hat{f}$ has learned something about f.

Variance Estimation The learner-PD and learner-PFI vary not only due to learner variance (refitted models), but also due to using different samples each time for the Monte Carlo integration. Therefore, their variance estimates capture the entire modeling process. Consequently, learner-PD/PFI along with their variance estimators bring us closer to the DGP-PD/PFI, and only the systematic bias remains unknown.

We can estimate this point-wise variance of the learner-PD with:

$$\widehat{\mathbb{V}}(\overline{\widehat{PD}}(x)) = \left(\frac{1}{m} + c\right) \cdot \frac{1}{(m-1)} \sum_{d=1}^{m} (\widehat{PD}_{\hat{f}_d}(x) - \overline{\widehat{PD}}(x))^2$$

And equivalently for the learner-PFI:

$$\widehat{\mathbb{V}}(\overline{\widehat{PFI}}) = \left(\frac{1}{m} + c\right) \cdot \frac{1}{(m-1)} \sum_{d=1}^{m} (\widehat{PFI}_{\hat{f}_d} - \overline{\widehat{PFI}})^2$$

The correction term c depends on the data setting. In simulation settings that allow us to draw new training and test sets for each model, we can use $c = 0$, yielding the standard variance estimators. In real world settings, we usually have a fixed dataset of size n, and models are refitted using resampling techniques. Consequently, data are shared by model refits, and variance estimators will underestimate the true variance [35]. To correct the variance estimate of the generalization error for bootstrapped or subsampled models, Nadeau and Bengio [35] suggested the correction term $c = \frac{n_2}{n_1}$ (where n_2 and n_1 are sizes of test and training data). However, the correction remains a rough correction, relying on the strongly simplifying assumption that the correlation between model refits depends only on the number of shared observations in the respective training datasets and not on the specific observations that they share. While this assumption is usually wrong, we show in Section 3.1 that the correction term offers a vast improvement for variance estimation – compared to using no correction.

Confidence Bands and Intervals Since the learner-PD and learner-PFI are means with estimated variance, we can use the t-distribution with $m-1$ degrees of freedom to construct confidence bands/intervals, where m is the number of model fits. The point-wise confidence band for the learner-PD is:

$$CI_{\overline{\widehat{PD}}(x)} = \left[\overline{\widehat{PD}}(x) \pm t_{1-\frac{\alpha}{2}} \sqrt{\hat{\mathbb{V}}(\overline{\widehat{PD}}(x))}\right],$$

where $t_{1-\frac{\alpha}{2}}$ is the respective $1-\alpha/2$ quantile of the t-distribution with $m-1$ degrees of freedom. Equivalently, we propose a confidence interval for the learner-PFI:

$$CI_{\overline{\widehat{PFI}}} = \left[\overline{\widehat{PFI}} \pm t_{1-\frac{\alpha}{2}} \sqrt{\hat{\mathbb{V}}(\overline{\widehat{PFI}})}\right].$$

Taking the learner variance into account can affect the interpretation, as we show in the application in Section 4. An additional advantage of the learner-PD and learner-PFI is that they make better use of the data, since a larger share of the data is employed as test data compared to only using a small holdout set.

3 Simulation Studies

In this Section, we study the coverage of the confidence intervals for the learner-PD/PFI on simulated examples (Section 3.1) and compare our proposed refitting-based variance estimation with model-based variance estimators (Section 3.2).

3.1 Confidence Interval Coverage Simulation

In simulations, we compared confidence interval performance between bootstrapping and subsampling, with and without variance correction. We simulated two DGPs: a *linear* DGP was defined as $y = f(x) = x_1 - x_2 + \epsilon$ and a *non-linear* DGP as $y = f(x) = x_1 - \sqrt{1 - x_2} + x_3 \cdot x_4 + (x_4/10)^2 + \epsilon$. All features were uniformly sampled from the unit interval $[0;1]$, and for both DGPs, we set $\epsilon \sim N(0,1)$. We studied the two settings "simulation" and "real world" as described in Section 2.1. In both settings, we trained linear models (lm), regression trees (tree) and random forests (rf) each 15 times, and computed confidence intervals for the learner-PD and learner-PFI across the 15 refitted models. In the "simulation" setting, we sampled $n \in \{100, 1000\}$ fresh data points for each model refit, where 63.2% of the data were used for training and the remaining 36.8% for PDP and PFI estimation.[14]

In the "real world" setting, we sampled $n \in \{100, 1000\}$ data points **once** per experiment, and generated 15 training data sets using a bootstrap (sample size

[14] We choose this training size (63.2%) to match the expected number of unique samples when using bootstrapping, which allows to compare bootstrapping and subsampling.

Table 1: Coverage Probability of the 95% Confidence Bands/Intervals for PDP and PFI. boot = bootstrap, subs = subsampling, * = with adjustment.

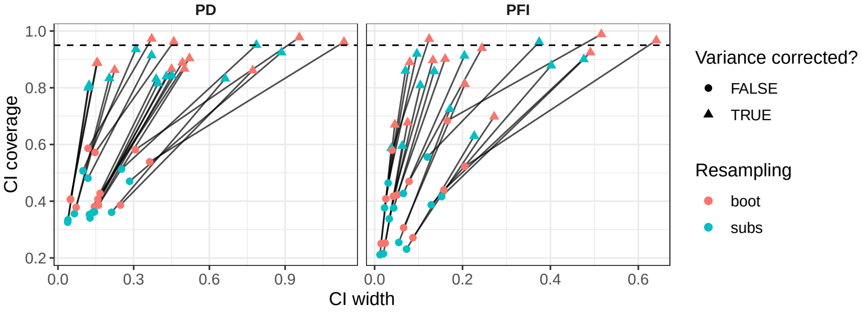
			PD					PFI				
dgp	model	n	boot	boot*	subs	subs*	ideal	boot	boot*	subs	subs*	ideal
linear	lm	100	0.41	0.89	0.34	0.82	0.95	0.27	0.70	0.23	0.63	0.94
linear	lm	1000	0.41	0.89	0.33	0.80	0.95	0.25	0.68	0.21	0.60	0.95
linear	rf	100	0.39	0.86	0.36	0.83	0.95	0.44	0.92	0.39	0.88	0.95
linear	rf	1000	0.38	0.87	0.35	0.83	0.95	0.42	0.90	0.38	0.86	0.95
linear	tree	100	0.54	0.96	0.47	0.92	0.95	0.52	0.97	0.42	0.90	0.95
linear	tree	1000	0.57	0.96	0.48	0.91	0.95	0.42	0.90	0.34	0.81	0.95
non-linear	lm	100	0.43	0.90	0.36	0.84	0.95	0.31	0.81	0.25	0.72	0.94
non-linear	lm	1000	0.41	0.89	0.33	0.81	0.95	0.25	0.67	0.21	0.59	0.95
non-linear	rf	100	0.39	0.87	0.36	0.84	0.95	0.47	0.94	0.43	0.91	0.95
non-linear	rf	1000	0.38	0.86	0.36	0.83	0.95	0.41	0.89	0.38	0.86	0.95
non-linear	tree	100	0.58	0.98	0.51	0.95	0.95	0.68	0.99	0.56	0.96	0.94
non-linear	tree	1000	0.59	0.97	0.51	0.94	0.95	0.58	0.97	0.46	0.92	0.95

n with replacement, which yields $0.632 \cdot n$ unique data points in expectation) or subsampling (sample size $0.632 \cdot n$ without replacement). In both settings, the learner-PD and learner-PFI as well as their respective confidence intervals were computed over the 15 retrained models. We repeated the experiment 10,000 times and counted how often the estimated confidence intervals covered the expected PD or PFI ($\mathbb{E}_F[PD_{\hat{f}}]$ and $\mathbb{E}_F[PFI_{\hat{f}}]$) over the distribution of models F.[15] These expected values were computed using 10,000 separate runs. The coverage estimates were averaged across features per scenario and for PD also across grid points ($\{0.1, 0.3, 0.5, 0.7, 0.9\}$) for all features.

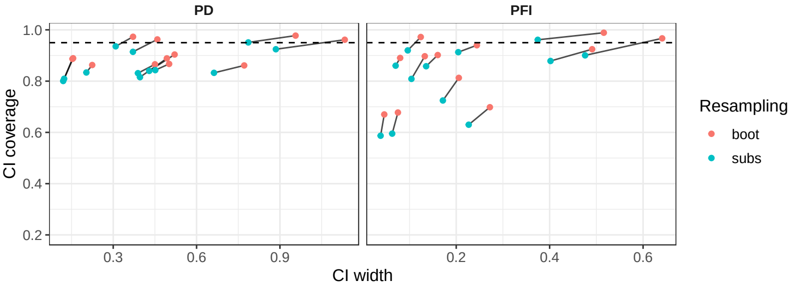
Table 1 shows that in the "simulation" setting ("ideal"), we can recover confidence intervals using the standard variance estimation with the desired coverage probability. However, in the "real world" setting, the confidence intervals for both the learner-PD and learner-PFI are too narrow across all scenarios and both resampling strategies when the intervals are based on naive variance estimates. Some coverage probabilities are especially low, such as for linear models with $30\% - 40\%$.

The coverage probabilities drastically improve when the correction term is used (see Figure 4a). However, in the simulated scenarios, these probabilities are still somewhat too narrow. For the linear model, the confidence intervals were the narrowest, with coverage probabilities of around $80\% - 90\%$ for PD and $60\% - 80\%$ for PFI across DGPs and sample sizes. The PD confidence bands were not heavily affected by increasing sample size n, but the PFI estimates became slightly narrower in most cases. In the case of decision trees, the ad-

[15] The coverage does not refer to the DGP-PD/PFI, but rather to the expected learner-PD/PFI, as we studied the choices of resampling and correction for the learner variance.



(a) CIs with vs without variance correction.



(b) Bootstrapping- vs subsampling-based CIs (with variance correction).

Figure 4: Confidence interval width vs. coverage for bootstrapping (boot) and subsampling (subs), segments connect identical scenarios.

justed confidence intervals were sometimes too large, especially for the adjusted bootstrap.

Except for trees on the *non-linear* DGP, the bootstrap outperformed subsampling in terms of coverage, i.e. the coverage was closer to the 95% level and rather erred on the side of "caution" with wider confidence intervals (see Figure 4b). As recommended by Nadeau and Bengio [35], we used 15 refits. We additionally analyzed how the coverage and interval width changed by increasing refits from 2 to 30 and noticed that the coverage worsened with more refits while the width of the confidence intervals decreased. Increasing the number of refits incurs an inherent trade-off between interval width and coverage: The more refits are considered, the more accurate the learner-PFI and learner-PD become, and also the more certain the variance estimates become, scaling with $1/m$. However, there is a limit to the information in the data, such that additional refits falsely reduce the variance estimate and the confidence intervals become too narrow. To refit the model 10 - 20 times seemed to be an acceptable trade-off between coverage and interval width, as demonstrated in Figure 5. Below $\sim$ 10 refits, the confidence intervals were large and the mean PD/PFI estimates have a high variance. Above ∼ 20 refits, the widths no longer decreased substantially. The figures for the other scenarios can be found in online Appendix I [31].[16] With our

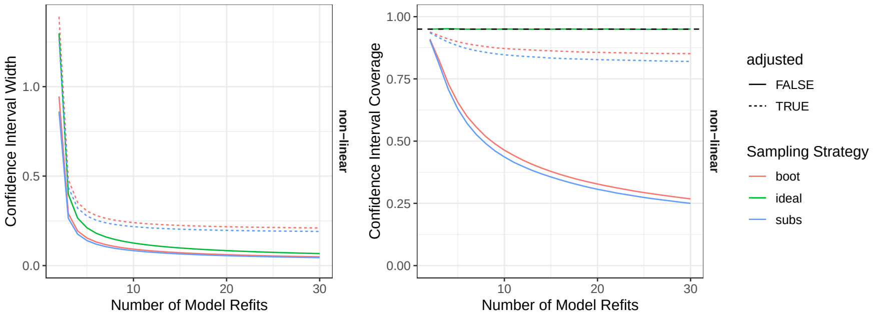


Figure 5: Average PD confidence band width (left) and coverage (right) as a function of the number of refitted models for the random forest on the *non-linear* DGP.

simulation results, we could show that employing confidence intervals using the naive variance estimation (without correction) results in considerably too narrow intervals. While the simple correction term by Nadeau and Bengio [35] does not always provide the desired coverage probability, it is a vast improvement over the naive approach. We therefore recommend using the correction when computing confidence intervals for learner-PD and learner-PFI, as this is currently the best approach available. We also recommend refitting the model approximately 15 times. For more "cautious" confidence intervals, we recommend using confidence intervals based on resampling with replacement (bootstrap) over sampling without replacement (subsampling). However, besides wider confidence intervals, the bootstrap also requires additional attention when model-tuning with internal resampling is used; otherwise, data points may inadvertently be used in both training and validation datasets.

3.2 Comparison to Model-based Approaches

While our methods based on model-refits provide confidence intervals for PD and PFI in a model-agnostic manner, it is also possible to exploit (co)variance estimates of probabilistic models to construct confidence intervals. Here, we will, for the case of PD[17], compare our approach with the model-based approach of

[16] The CI coverage and width: for PD with n=100 can be found in Figure I.1 and Figure I.2; for PD with n=1000 can be found in Figure I.3 and Figure I.4; for PFI with n=100 can be found in Figure I.5 and Figure I.6; for PFI with n=1000 can be found in Figure I.7 and Figure I.8.

[17] We do not know of any application of Moosbauer et al.'s [34] approach to PFI of probabilistic models.

Table 2: Coverage probabilities for 95% confidence bands of PD estimates for model-based (mod) and subsampling-based (subs) approaches. Results are averaged over all features and grid points for the GP and LM. The experiments were conducted on two different sample sizes n. Furthermore, mean (standard deviation) of confidence width are reported for both approaches. The last column contains the standard deviation of the MC error for the model-based approach.

dgp	model	n	coverage		width (sd)		
			mod	subs	mod	subs	mod
1	gp	200	0.66	0.95	0.36 (0.19)	0.48 (0.11)	0.15
1	gp	1000	0.71	0.97	0.28 (0.31)	0.24 (0.07)	0.07
1	lm	200	0.34	0.95	0.15 (0.03)	0.41 (0.10)	0.15
1	lm	1000	0.35	0.95	0.06 (0.01)	0.19 (0.05)	0.07

Moosbauer et al. [34] applied to a Gaussian Process (GP) and a linear model (LM).[18] We find that our approach more reliably delivers better coverages that are closer to the $1-\alpha$ confidence level; this can be explained by the fact that the model-based approach ignores the variance in Monte Carlo integration.

We consider the following simulation setting:

$$\text{DGP:} \quad Y = 4X_1 - 2X_2 + 2X_3 - X_4 + X_5 + \epsilon$$

with $X_j \overset{i.i.d.}{\sim} U(0,1)$ for all $j \in \{1,...,5\}$. Given a DGP of the form $y = f(x) + \epsilon$ the distribution of ϵ is set to $\epsilon \sim N(0, (0.2\,\sigma(f(x)))^2)$.

We calculate the DGP-PD analytically. The experiments are performed 1000 times for $n = 200$ and $n = 1000$, where a random sample of $n_1 = 0.632 \cdot n$ is used to fit the models and the remaining $n_2 = 0.368 \cdot n$ observations are used to calculate the PD. Since model-based variance estimates for linear models can be derived analytically based on the variance of their coefficients, we additionally compare these estimates to our resampling-based approach (i.e. the learner-PD) for a correctly specified linear model. The model-based variance estimates can be calculated by one model fit per repetition. In contrast, we use 15 refits on subsampled data sets per repetition to compute the variance estimate for the resampling-based approach.[19][20] We choose the grid points $\{0.1, 0.3, 0.5, 0.7, 0.9\}$ and a confidence level of 0.95 to evaluate the mean and variance estimates of the PDs. Table 2 shows the results for both the model-based (mod) and the adjusted subsampling-based (subs) approach. While the subsampling-based approach shows almost perfect coverages for the different settings, the model-based

[18] More details on the approach of Moosbauer et al. [34] are provided in online Appendix J [31].

[19] We use a marginal sampler for perturbations (since we assume uncorrelated features in all scenarios).

[20] We did not consider the bootstrapping approach in our experiments as we encountered numerical issues in the invertability of the covariance matrix (due to duplicated values introduced by bootstrap) [42].

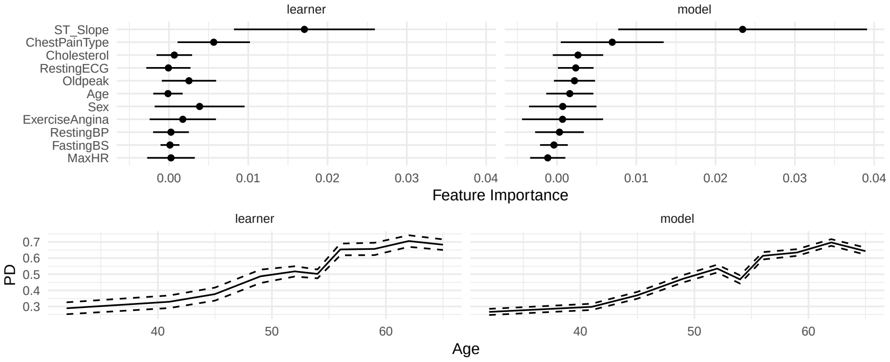


Figure 6: Top: Conditional Learner-PFI and model-PFI with point-wise 95%-confidence intervals for the random forest. Bottom: Conditional Learner-PDP and model-PDP with point-wise 95%-confidence bands for the random forest and feature `Age`.

approach is far off the nominal level with values around 0.35 for the correctly specified linear model. This gap can be explained by the MC integration variance which is not incorporated in the model-based approaches. Hence, if the MC error is relatively high compared to the model variance, coverages are bad. To illustrate this relationship, we calculated the average standard deviation of the MC integration variance estimator (see Eq. (5)) for the model-based approaches (see Table 2). Since the confidence bands of these approaches only cover the model variance, the confidence width is directly proportional to the model variance. If we compare the "MC se" column with the average widths of the model-based approach, it is observable that coverages are rather low (e.g., 0.34 for LM with $n = 200$) in the case where "MC se" divided by width is rather high (e.g., $0.15/0.15 = 1$) and vice versa.

Thus, if the main goal is to quantify both uncertainty sources inherent in the PD estimation and thus to receive reasonable coverages, the model-based approach cannot be recommended since only one of two sources of variability are covered by the estimates. Even for the linear model, which is commonly used for inferential purposes, the confidence bands for the PD estimates might be far too conservative as shown in Table 2. The subsampling-based variance estimates we proposed in this work however cover both the learner variance and the MC error and provide satisfying coverage values.

4 Application

We apply our proposed estimators to the motivational example from Section 1.1. We supposed that a researcher predicted chronic heart disease [15] ($n = 918$) from sociological and medical indicators such as age, blood pressure and maximum heart rate. She fitted one random forest and estimated conditional PFI and conditional PDPs to interpret the result.

Instead of only computing the conditional PFI and conditional PDP for one model, we estimate the proposed conditional model-PFI and conditional learner-PFI along with the proposed confidence intervals. For the learner-based insights, we therefore refitted the model 15 times on resampled training sets.

Figure 6 shows model and learner based conditional PFI and conditional PDP with the corresponding confidence intervals ($\alpha = 0.05$).

Learner-PFI and model-PFI disagree on the ordering of the features: they agree that slope of the ECG segment (`STSlope`) and the type of chest pain (`ChestPainType`) are the most important features; but learner-PFI ranks sex (`Sex`) and ST depression induced by exercise relative to rest (`Oldpeak`) next, while model-PFI ranks cholesterol (`Cholesterol`) second and resting state ECG (`RestingECG`) third. For both model-PFI and learner-PFI all except two confidence intervals include zero, namely `STSlope` and `ChestPainType`. The confidence intervals for model-PFI and learner-PFI indicate that both learner variance and the uncertainty stemming from the Monte Carlo integration are relatively high. The model-PFI cannot tell us to what extent the estimate varies due to learner variance; only the learner-PFI can quantify the learner variance.

Figure 6, bottom row, shows both the conditional model-PDP and the conditional learner-PDP for age (`Age`). Model-PDP and learner-PDP agree that individuals of higher age are more likely to have heart disease with a strong increase in prevalence around the age of 55. However, the confidence bands of the learner-PDP are wider than those of the model-PDP. Furthermore, the bump that can be observed in the model-PDP around the age of 50 is smoother in the learner PDP and should partly be attributed to uncertainties involved in model fitting. Neglecting the learner variance would mean being overconfident about the partial dependence curve. In particular, the Monte Carlo approximation error decreases with $1/n$ as the sample size n for PD and PFI estimation increases. Wrongly interpreted, this can lead to a false sense of confidence in the estimated effects and importance since only one model is considered and learner variance is ignored.

5 Discussion

We related the PD and the PFI to the DGP, proposed variance and confidence intervals, and discussed conditions for inference. Our derivations were motivated by taking an external view of the statistical inference process and postulating that there is a ground truth counterpart to PD/PFI in the DGP. To the best of our knowledge, statistical inference via model-agnostic interpretable machine learning is already used in practice, but under-explored in theory.

A critical assumption for inference of effects and importance using interpretable machine learning is the unbiasedness of the learner. The learner bias is difficult to test, and can be introduced by e.g. choice of model class, regularization, and feature selection. For example, regularization techniques such as

LASSO introduce a small bias *on purpose* [44] to decrease learner variance and improve predictive performance. We must better understand how specific biases affect the prediction function and consequently PD and PFI estimates.

Another crucial limitation for inference of PD and PFI is the underestimation of variance due to data sharing between model refits. While we could show that a simple correction of the variance [35] vastly improves the coverage, a proper estimation of the variance remains an open issue. A promising approach relying on repeated nested cross validation to correctly estimate the variance was recently proposed by Bates et al. [6]. However, this approach is more computationally intensive by a factor of up to 1,000.

Furthermore, samplers are not readily available. Especially conditional sampling is a complex problem, and samplers must be trained using data. Training samplers even introduces another source of uncertainty to our estimates that we neglected in our work. It is difficult to separate this source of uncertainty from the uncertainty of the model learner, since trained samplers are correlated not only with each other, but possibly also with the trained models. We see integrating sampler uncertainty as an important step in providing reliable uncertainty estimates in practice, but we leave this to future work.

Statements and Declarations

Funding This project is supported by the Bavarian State Ministry of Science and the Arts, the Bavarian Research Institute for Digital Transformation (bidt), the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A, by the German Research Foundation (DFG) – Emmy Noether Grant 437611051 to MNW, and the Carl Zeiss Foundation (project on "Certification and Foundations of Safe Machine Learning Systems in Healthcare"). The authors of this work take full responsibilities for its content.

Availability of Data, Code, and Online Appendix The data used in the application is openly available and referenced in this paper. The code for visualizations, simulations and the application is written in the R programming language [40] and is publicly available via `https://github.com/gcskoenig/paper_inference_code`. The online Appendix is available via [31].

References

1. Altmann, A., Toloşi, L., Sander, O., Lengauer, T.: Permutation importance: a corrected feature importance measure. Bioinformatics **26**(10), 1340–1347 (2010)
2. Apley, D.W., Zhu, J.: Visualizing the effects of predictor variables in black box supervised learning models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **82**(4), 1059–1086 (2020)
3. Archer, K.J., Kimes, R.V.: Empirical characterization of random forest variable importance measures. Computational Statistics & Data Analysis **52**(4), 2249–2260 (2008)
4. Bair, E., Ohrbach, R., Fillingim, R.B., Greenspan, J.D., Dubner, R., Diatchenko, L., Helgeson, E., Knott, C., Maixner, W., Slade, G.D.: Multivariable modeling of phenotypic risk factors for first-onset tmd: the oppera prospective cohort study. The Journal of Pain **14**(12), T102–T115 (2013)
5. Bates, S., Candès, E., Janson, L., Wang, W.: Metropolized knockoff sampling. Journal of the American Statistical Association **116**(535), 1413–1427 (2021)
6. Bates, S., Hastie, T., Tibshirani, R.: Cross-validation: what does it estimate and how well does it do it? Journal of the American Statistical Association pp. 1–12 (2023)
7. Blesch, K., Watson, D.S., Wright, M.N.: Conditional feature importance for mixed data. AStA Advances in Statistical Analysis pp. 1–20 (2023)
8. Boulesteix, A.L., Wright, M.N., Hoffmann, S., König, I.R.: Statistical learning approaches in the genetic epidemiology of complex diseases. Human Genetics **139**(1), 73–84 (2020)
9. Breiman, L.: Random forests. Machine learning **45**(1), 5–32 (2001)
10. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.: Classification and regression trees. CRC Press (1984)
11. Cafri, G., Bailey, B.A.: Understanding variable effects from black box prediction: Quantifying effects in tree ensembles using partial dependence. Journal of Data Science **14**(1), 67–95 (2016)
12. Candes, E., Fan, Y., Janson, L., Lv, J.: Panning for gold: 'model-x'knockoffs for high dimensional controlled variable selection. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **80**(3), 551–577 (2018)
13. Chen, H., Janizek, J.D., Lundberg, S., Lee, S.I.: True to the model or true to the data? arXiv preprint arXiv:2006.16234 (2020)
14. Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., Robins, J.: Double/debiased machine learning for treatment and structural parameters. The Econometrics Journal **21**(1), C1–C68 (2018)
15. Dua, D., Graff, C.: UCI machine learning repository (2017), `http://archive.ics.uci.edu/ml`
16. Emrich, E., Pierdzioch, C.: Public goods, private consumption, and human capital: Using boosted regression trees to model volunteer labour supply. Review of Economics/Jahrbuch für Wirtschaftswissenschaften **67**(3) (2016)
17. Esselman, P.C., Stevenson, R.J., Lupi, F., Riseng, C.M., Wiley, M.J.: Landscape prediction and mapping of game fish biomass, an ecosystem service of michigan rivers. North American Journal of Fisheries Management **35**(2), 302–320 (2015)
18. Fisher, A., Rudin, C., Dominici, F.: All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research **20**(177), 1–81 (2019)

19. Freiesleben, T., König, G., Molnar, C., Tejero-Cantero, A.: Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena. arXiv preprint arXiv:2206.05487 (2022)
20. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Annals of statistics pp. 1189–1232 (2001)
21. Geman, S., Bienenstock, E., Doursat, R.: Neural networks and the bias/variance dilemma. Neural Computation **4**(1), 1–58 (1992)
22. Grange, S.K., Carslaw, D.C.: Using meteorological normalisation to detect interventions in air quality time series. Science of The Total Environment **653**, 578–588 (2019)
23. Groemping, U.: Model-agnostic effects plots for interpreting machine learning models. Reports in Mathematics, Physics and Chemistry, Department II, Beuth University of Applied Sciences Berlin. **Report 1/2020** (2020)
24. Hastie, T., Tibshirani, R., Friedman, J.H., Friedman, J.H.: The elements of statistical learning: data mining, inference, and prediction, vol. 2. Springer (2009)
25. Hooker, G., Mentch, L., Zhou, S.: Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. Statistics and Computing **31**, 1–16 (2021)
26. Ishwaran, H., Lu, M.: Standard errors and confidence intervals for variable importance in random forest regression, classification, and survival. Statistics in Medicine **38**(4), 558–582 (2019)
27. Janitza, S., Celik, E., Boulesteix, A.L.: A computationally fast variable importance test for random forests for high-dimensional data. Advances in Data Analysis and Classification **12**(4), 885–915 (2018)
28. König, G., Molnar, C., Bischl, B., Grosse-Wentrup, M.: Relative feature importance. In: 2020 25th International Conference on Pattern Recognition (ICPR). pp. 9318–9325. IEEE (2021)
29. van der Laan, M.J., Rose, S., Zheng, W., van der Laan, M.J.: Cross-validated targeted minimum-loss-based estimation. Targeted learning: causal inference for observational and experimental data pp. 459–474 (2011)
30. Mitchell, T.M.: The need for biases in learning generalizations. Citeseer (1980)
31. Molnar, C., Freiesleben, T., König, G., Herbinger, J., Reisinger, T., Casalicchio, G., Wright, M.N., Bischl, B.: Online appendix for "Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process" (6 2023). `https://doi.org/10.6084/m9.figshare.23294945.v1`
32. Molnar, C., König, G., Bischl, B., Casalicchio, G.: Model-agnostic feature importance and effects with dependent features: a conditional subgroup approach. Data Mining and Knowledge Discovery pp. 1–39 (2023)
33. Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C.A., Casalicchio, G., Grosse-Wentrup, M., Bischl, B.: General pitfalls of model-agnostic interpretation methods for machine learning models, pp. 39–68. Springer International Publishing (2022)
34. Moosbauer, J., Herbinger, J., Casalicchio, G., Lindauer, M., Bischl, B.: Explaining hyperparameter optimization via partial dependence plots. Advances in Neural Information Processing Systems **34**, 2280–2291 (2021)
35. Nadeau, C., Bengio, Y.: Inference for the generalization error. Machine Learning **52**(3), 239–281 (2003)
36. Obringer, R., Nateghi, R.: Predicting urban reservoir levels using statistical learning techniques. Scientific Reports **8**(1), 1–9 (2018)

37. Page, W.G., Wagenbrenner, N.S., Butler, B.W., Forthofer, J.M., Gibson, C.: An evaluation of ndfd weather forecasts for wildland fire behavior prediction. Weather and Forecasting **33**(1), 301–315 (2018)
38. Parr, T., Wilson, J.D.: A stratification approach to partial dependence for codependent variables. arXiv preprint arXiv:1907.06698 (2019)
39. Parr, T., Wilson, J.D., Hamrick, J.: Nonparametric feature impact and importance. arXiv preprint arXiv:2006.04750 (2020)
40. R Core Team: R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria (2018), `https://www.R-project.org/`
41. Ribeiro, M.T., Singh, S., Guestrin, C.: Model-agnostic interpretability of machine learning. ICML WHI '16 (2016), arXiv preprint arXiv:1606.05386
42. Roustant, O., Ginsbourger, D., Deville, Y.: Dicekriging, diceoptim: Two r packages for the analysis of computer experiments by kriging-based metamodeling and optimization. Journal of Statistical Software **51**(1), 1–55 (2012)
43. Stachl, C., Au, Q., Schoedel, R., Gosling, S.D., Harari, G.M., Buschek, D., Völkel, S.T., Schuwerk, T., Oldemeier, M., Ullmann, T., Hussmann, H., Bischl, B., Bühner, M.: Predicting personality from patterns of behavior collected with smartphones. Proceedings of the National Academy of Sciences **117**(30), 17680–17687 (2020)
44. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological) **58**(1), 267–288 (1996)
45. Watson, D.S., Wright, M.N.: Testing conditional independence in supervised learning algorithms. Machine Learning **110**, 2107–2129 (2021)
46. Williamson, B.D., Gilbert, P.B., Carone, M., Simon, N.: Nonparametric variable importance assessment using machine learning techniques. Biometrics (2019)
47. Williamson, B.D., Gilbert, P.B., Simon, N.R., Carone, M.: A general framework for inference on algorithm-agnostic variable importance. Journal of the American Statistical Association pp. 1–14 (2021)
48. Zhang, L., Janson, L.: Floodgate: inference for model-free variable importance. arXiv preprint arXiv:2007.01283 (2020)
49. Zhao, Q., Hastie, T.: Causal interpretations of black-box models. Journal of Business & Economic Statistics **39**(1), 272–281 (2021)

2.4 Paper IV: Relative Feature Importance

König, Gunnar, Christoph Molnar, Bernd Bischl and Moritz Grosse-Wentrup. **Relative Feature Importance (RFI).** *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021.

Gunnar König contributed to the paper as first author. Gunnar König had the initial idea and wrote large parts of the paper. Christoph Molnar partly wrote the section on estimation and testing, reviewed the software code, and extensively proofread the mathematical proofs. All authors added input, suggested modifications proofread and revised the paper.

Relative Feature Importance

Gunnar König[1,2], Christoph Molnar[1], Bernd Bischl[1], Moritz Grosse-Wentrup[2,3,4]
[1]Institute for Statistics, LMU Munich, [2]Research Group Neuroinformatics, University of Vienna,
[3]Research Platform Data Science @ Uni Vienna, [4]Vienna Cognitive Science Hub

Abstract—**Interpretable Machine Learning (IML) methods are used to gain insight into the relevance of a feature of interest for the performance of a model. Commonly used IML methods differ in whether they consider features of interest in isolation, e.g., Permutation Feature Importance (PFI), or in relation to all remaining feature variables, e.g., Conditional Feature Importance (CFI). As such, the perturbation mechanisms inherent to PFI and CFI represent extreme reference points. We introduce Relative Feature Importance (RFI), a generalization of PFI and CFI that allows for a more nuanced feature importance computation beyond the PFI versus CFI dichotomy. With RFI, the importance of a feature relative to any other subset of features can be assessed, including variables that were not available at training time. We derive general interpretation rules for RFI based on a detailed theoretical analysis of the implications of relative feature relevance, and demonstrate the method's usefulness on simulated examples.**

Index Terms—**feature importance, interpretable machine learning, explainable artificial intelligence, causality**

I. Introduction

Predictive modelling is increasingly deployed in high-stakes environments, e.g., in the criminal justice system [11], loan approval [32], recruiting [9] and medicine [27]. Due to legal regulations [10], [29] and ethical considerations, ML methods need not only perform robustly in such environments but also be able to justify their recommendations in a human-intelligible fashion. This development has given rise to the field of interpretable machine learning (IML) that involves studying methods that provide insight into the relevance of features for model performance, referred to as feature importance.

Prominent feature importance techniques include permutation feature importance (PFI) [5], [12] and conditional feature importance (CFI) [12], [19], [25]. PFI is based on replacing the feature of interest X_j with a perturbed version sampled from the marginal distribution $P(X_j)$ while CFI perturbs X_j such that the conditional distribution with respect to the set R of remaining features $P(X_j|X_R)$ is preserved. The sampling strategy defines the method's reference point and therefore affects the method's implicit notion of relevance. While PFI quantifies the overall reliance of the model on the feature of interest, CFI quantifies its unique contribution given *all* remaining features.

While both PFI and CFI are useful, they fail to answer more nuanced questions of feature importance. For instance, a stakeholder may be interested in the importance of a feature relative to a subset of features. Also, the user may want to know how important a feature is relative to variables that had not been available at training time.

We suggest relative feature importance (RFI) as a generalization of PFI and CFI that moves beyond the dichotomy between PFI, which breaks all dependencies with features, and CFI, which preserves all dependencies with features. In contrast to PFI and CFI, RFI is based on a perturbation that is restricted to preserve the relationships with a set of variables G *that can be chosen arbitrarily*. We show that RFI is (1) semantically meaningful and (2) practically useful.

We demonstrate the semantical meaning of RFI in Section IV. In particular, we derive general interpretation rules that link nonzero RFI to (1) the conditional dependence of the feature of interest with the target and non-conditioned features $X_{\underline{R}}$ given the conditioned variables X_G in the data and (2) the conditional dependence of the input to the feature of interest X_j with the model's prediction $\hat{Y}$ given fixed inputs to the remaining features X_R (Theorem 1). Furthermore, we show that a nonzero difference between RFI_j^G and $\mathrm{RFI}_j^{G \cup N}$, with N being an arbitrary set disjunct with G, implies the conditional dependence $X_j \not\perp\!\!\!\perp X_N|X_G$ (Theorem 2).

In Section V, we provide an implementation of RFI estimation that is based on recent results from the related knockoff research field [7], [23]. Furthermore, we translate the testing framework developed for conditional feature importance [30] to RFI. We support our theoretical analysis and findings by various simulation studies in Section VI. In particular, we show that RFI can expose the indirect contribution of variables that are not directly used by the model but provide information via dependent variables (Section VI-A). Similarly, we show how RFI can be used to assess feature importance with respect to variables not included at training time (Section VI-B).

A. Contributions and Related Work

While conditioning on subsets of variables has been suggested before [12], [25], the implications of this generalized variant of CFI have not yet been rigorously analyzed. Some IML methods perturb or hide subsets

This work is funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A and supported by the Bavarian State Ministry of Science and the Arts in the framework of the Centre Digitisation.Bavaria (ZD.B). The authors of this work take full responsibility for its content.

of features, e.g., in the context of multiple regression relative importance analysis is a model-specific technique that averages over all importances of models trained on feature subsets [6], [16]. Model-agnostic, local approximations to the respective feature effect that avoid retraining and instead perturb subsets of features have also been proposed [17], [33]. A very recent global, model-agnostic feature importance proposal called SAGE quantifies feature importance by perturbing multiple features [8].

While the aforementioned approaches are all based on removing several features to provide more nuanced insight into the model, our proposal only modifies the feature of interest. Our approach is model-agnostic and global, while most aforementioned approaches are model-specific or local. The exception is the global, model-agnostic SAGE [8], however the approaches are not only computationally but also semantically different. E.g. our method assigns an importance of zero for features that are not used by the model[1], which is not the case for SAGE. While our approach aims to provide nuanced insights into variable importance relative to a specific set, SAGE aims to quantify the overall importance of variables for the model.

Feature importance relative to variables that have not been included in the training set has not been studied before. The indirect influence of variables that the model does not computationally rely but statistically depend on has been studied e.g. in [1].

II. Background and Notation

A. Notation

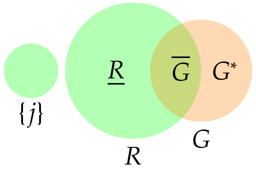


Fig. 1. Overview of our notation.

We denote the target variable, i.e., the variable the model predicts, as Y and feature variables by $X_{(.)}$. We refer to the variables as features to emphasize when they were used in model training. Their observations are denoted by y and $x_{(.)}$. We use $D := \{1, \dots, p\}$ for the index set of all features included in model training and j for the index of our feature of interest, X_j. The index set of the remaining variables is denoted as $R := D \backslash \{j\}$ (rest, remainder). The index set of features, relative to which the importance of X_j is considered, is denoted as G. As G can refer to any index set of variables, we denote its intersection with R as $\overline{G} = R \cap G$ and its complement as $\underline{R} = R \backslash G$. We denote the index set of conditioning variables that were not made available to the model during training as $G^* = G \backslash R$.

In case we add new elements to the conditioning set G, we will denote this set as N. The set may include variables within and outside D. The respective components are denoted as $N^* = N \backslash R$ and as $\overline{N} = R \cap N$. The remainder of R without G and N is denoted as $\underline{\underline{R}} = \underline{R} \backslash N$. We denote perturbed variables of interest relative to G as $\tilde{X}_j^G$. We refer to the original and perturbed probability distribution of X_j as the observational and interventional distribution $P(X_j, \dots)$ and $P(\tilde{X}_j^G, \dots)$. The inspected model is denoted as f, its prediction as $\hat{Y}$. Independence of Y and X conditional on Z is denoted using $X \perp\!\!\!\perp Y | Z$, the respective conditional dependence as $X \not\perp\!\!\!\perp Y | Z$.

B. Feature Importance

Performance-based feature importance methods assess the relevance of a feature of interest X_j by assessing the impact of a perturbation of X_j on the model's performance. Local feature importance methods focus on the importance of features for specific data points, whereas global feature importance methods assess the impact over the whole domain. In the following, we focus on global methods.

Global feature importance is computed according to the following general schemata:

$$\mathrm{FI}_j = \tilde{\mathcal{R}}^j - \mathcal{R} \text{ or } \mathrm{FI}_j = \frac{\tilde{\mathcal{R}}^j}{\mathcal{R}}$$

where we denote the original risk of the model and the risk after perturbing X_j as $\mathcal{R}$ and $\tilde{\mathcal{R}}^j$, respectively. For estimation, the true risk $\mathcal{R}$ is replaced with the empirical risk $\mathcal{R}_{\mathrm{emp}}$.

Feature importance methods furthermore differ in how they perturb and whether they rely on retraining the model. While some methods retrain the model after the perturbation (e.g. LOCO, [15]), others evaluate the impact of the perturbation on the same original model (e.g. [5], [25]). In this work, we focus on methods that avoid retraining.

For methods that avoid retraining, we observe a dichotomy between two general perturbation approaches: resampling that preserves the *marginal* and resampling that preserves the *conditional* distribution. Marginal resampling was originally proposed to compute perturbed versions of X_j by permuting the observations $x_j^{(i)}$ within the sample [5]. The respective sample breaks the dependence between X_j and (Y, X_R) while preserving the marginal distribution $P(X_j)$. More recently, Model Reliance was proposed [12], which takes the expectation over all possible permutations. Resampling from the marginal distribution has been criticized to introduce bias, in particular because it overestimates the importance of correlated variables

[1] A proof of this property is given in Lemma 2.

[25], resulting in incorrect feature rankings [26]. It also leads to extrapolation under dependent features [14], [19], i.e. conclusions about the model are being drawn using unrealistic data points on which the model was not trained. CFI, on the other hand, samples from the conditional distribution $P(X_j|X_R)$ [2], [7], [12], [14], [19], [25], [28]. A large variety of model-specific methods exist [13], [31]. Conditional variants quantify the importance of a feature given the information that all remaining features R contain about X_j [20], thereby avoiding evaluation of the model on unrealistic datapoints [19].

III. Relative Feature Importance

Relative Feature Importance is a general framework that assesses feature importance relative to arbitrary variable sets G. The frameworks subsumes PFI and CFI as two extreme special cases.

In PFI, X_j is replaced with a perturbed version that preserves the marginal distribution $P(X_j)$ while breaking the dependencies with Y and all features. In CFI, a perturbed version of X_j is used that preserves the conditional distribution $P(X_j|X_R)$, thereby only breaking conditional dependence between X_j and Y given all features. As our analysis in Section IV establishes, the replacement strategies of PFI and CFI define extreme reference points. CFI quantifies the contribution relative to *all* remaining features R, whereas PFI regards a feature in isolation.

We go beyond the PFI versus CFI dichotomy. We argue that it is (1) meaningful (Section IV) and (2) practically useful (Section VI) to replace X_j with perturbed versions that preserve the conditional distribution $P(X_j|X_G)$ with respect to *arbitrary* sets G while requiring $\tilde{X}_j^G \perp\!\!\!\perp (X_{\underline{R}}, Y)|X_G$. G can be a subset of R, but can also include variables not available at training time such that $G \backslash R \neq \emptyset$. We term the resulting method Relative Feature Importance (RFI):

Definition 1 (Relative Feature Importance – RFI): We define Relative Feature Importance with respect to a feature set G with $Y \notin G$ and a fixed model f as

$$\mathrm{RFI}_j^G := \tilde{\mathcal{R}}^{j|G} - \mathcal{R},$$

where $\tilde{\mathcal{R}}^{j|G} := \mathcal{R}(Y, f(X_R, \tilde{X}_j^G))$ is the risk w.r.t. to a replacement variable $\tilde{X}_j^G$ and $\mathcal{R} = \mathcal{R}(Y, f(X_j, X_R))$ refers to the original risk. The replacement variable has to satisfy

- $\tilde{X}_j^G \sim P(X_j|X_G)$ and
- $\tilde{X}_j^G \perp\!\!\!\perp (X_{\underline{R}}, Y)|X_G$.

In the following section, we discuss the semantic meaning of RFI. The estimation of RFI is discussed in Section V.

IV. Interpreting Relative Feature Importance

IML techniques aim to provide insight into the model and, possibly, into the underlying data generating mechanism. However, IML techniques themselves are subject to interpretation. The characterization of an IML method by its mathematical definition is computationally precise, but has limited aid in guiding users to make conclusions about the underlying model and data. In this section we provide a (non-comprehensive) list of interpretation rules for RFI, that *characterize the method by how it behaves in its context*. This context includes *both the model and the underlying data generating mechanism*. More specifically, we link RFI to (conditional) independence in the underlying data set as well as to whether the model's prediction $\hat{Y}$ is constant in the argument x_j for a fixed x_R. While RFI can be used for quantification of feature importance, we focus our analysis on relevance as a binary property and characterize relative feature relevance (RFI $\neq 0$). We show that the implicit notion of relevance of RFI is defined by the choice of G. By modifying the conditioning set G beyond the PFI versus CFI dichotomy, we are able to gain insight into more nuanced aspects of the model and the data generating mechanism. The main results are given in Theorem 1 and Theorem 2. Furthermore, we highlight limitations stemming from the choice of the loss function L and the model fit for the interpretation, which are, in our humble opinion, underrepresented in the current discussion.

We structure our analysis by taking the user's perspective and asking "What can we infer from relative feature relevance?".

A. Implications of Relative Feature Relevance

In the following, we analyze the implications of RFI without further assumptions about model and data. We thereby distinguish between two levels of explanation. Relative feature relevance provides insight, both into *model* and *data*.

Theorem 1: If $RFI_j^G \neq 0$ then

- $X_j \not\!\perp\!\!\!\perp (Y, X_{\underline{R}})|X_G$ in the underlying distribution (data level)
- $\tilde{X}_j \not\!\perp\!\!\!\perp \hat{Y}|X_R$ w.r.t. the interventional distribution $P(X_j|X_G)P(X_G, X_{\underline{R}}) > 0$ (model level)

We prove Theorem 1 in two steps. First, we assess the implications of the respective independence for the underlying data set (Lemma 1). Then, we assess the implications of the respective independence for the model (Lemma 2). The contrapositions yield Theorem 1.

Lemma 1: If $X_j \perp\!\!\!\perp (Y, X_{\underline{R}})|X_G$ for any G with $Y \notin G$ then $RFI_j^G = 0$.

We base the proof of Lemma 1 on the insight that (because the model f is fixed) an equivalence in distribution implies an equivalence in risk (Proposition 1). Therefore conditions under which the interventional distribution $P(\tilde{X}_j^G, X_R, Y)$ coincides with the original distribution $P(X_j, X_R, Y)$ are sufficient for $RFI = 0$.

Proposition 1: If observational and interventional distribution coincide, then risks with and without perturbation are equal:

$$P(Y, X_j, X_R) = P(Y, \tilde{X}_j^G, X_R) \Rightarrow \mathcal{R}(f) = \tilde{\mathcal{R}}^{j|G}(f)$$

Proof of Proposition 1: Given that $P(Y, X_j, X_R) = P(Y, \tilde{X}_j, X_R)$ we can write

$$\begin{aligned}\mathcal{R}(f) &= \mathbb{E}_{Y,X_j,X_R}[L(Y, f(X_j, X_R))]\\ &= \mathbb{E}_{Y,\tilde{X}_j,X_R}[L(Y, f(\tilde{X}_j, X_R))] = \tilde{\mathcal{R}}(f).\end{aligned}$$

■

We show next that the conditional independence $X_j \perp\!\!\!\perp (X_{\underline{R}}, Y)|X_G$ is a sufficient condition for identity of both distributions.

Proof of Lemma 1: It holds that

$$\begin{aligned}P(Y, X_j, X_{\underline{R}}, X_G) &= P(X_j|Y, X_{\underline{R}}, X_G)P(Y, X_{\underline{R}}, X_G)\\ &\overset{X_j \perp\!\!\!\perp (X_{\underline{R}}, Y)|X_G}{=} P(X_j|X_G)P(Y, X_{\underline{R}}, X_G)\\ &\overset{\text{(def)}}{=} P(\tilde{X}_j^G|X_G)P(Y, X_{\underline{R}}, X_G)\\ &= P(\tilde{X}_j^G, Y, X_{\underline{R}}, X_G).\end{aligned}$$

Using Proposition 1 we can infer that $RFI_j^G = 0$.

■

So far, we have assessed implications for the underlying data generating mechanism. Next, we assess implications for the inspected model f.

Lemma 2: If $\tilde{X}_j^G \perp\!\!\!\perp \hat{Y}|X_R$ w.r.t. the interventional distribution $P(\tilde{X}_j^G, X_G, X_{\underline{R}})$ then $\text{RFI}_j^G = 0$ for any G.

Proof of Lemma 2: If the prediction for an observation $(x_1, \ldots, x_p)$ is independent of the value x'_j w.r.t. the interventional distribution, the prediction is unaffected when replacing x_j with any value x'_j with $P(x'_j|X_G = x_G)P(X_G = x_G, X_{\underline{R}} = x_{\underline{R}}) > 0$. Consequently, any sample from $\tilde{X}_j^G$ yields the same prediction.
Furthermore values x'_j with nonzero probability over the interventional distribution also have nonzero probability over the observational distribution. The interventional distribution can be rewritten as

$$\begin{aligned}P(\tilde{X}_j^G, X_G, X_{\underline{R}}) &= P(\tilde{X}_j^G|X_G, X_{\underline{R}})P(X_G, X_{\underline{R}})\\ &= P(\tilde{X}_j^G|X_G)P(X_G, X_{\underline{R}})\\ &= P(X_j|X_G)P(X_G, X_{\underline{R}}).\end{aligned}$$

Similarly, the observational distribution can be factorized into $P(X_j|X_G, X_{\underline{R}})P(X_G, X_{\underline{R}})$. As $P(X_j|X_G, X_{\underline{R}}) > 0 \Rightarrow P(X_j|X_G) > 0$ (which can be derived from, e.g., the law of total probability) it follows that $P(\tilde{X}_j^G, X_G, X_{\underline{R}}) > 0 \Rightarrow P(X_j, X_G, X_{\underline{R}}) > 0$.
Consequently the prediction $\hat{y}$ for any value x_j with positive probability $P(X_j = x_j|X_R = x_R)$ is identical given unchanged x_R.
As the conditional distributions of X_j and $\tilde{X}_j^G$ overlap and the distribution of X_R is unaffected, the prediction $\hat{Y}$ is identical with and without perturbation. Therefore $\mathcal{R} = \tilde{\mathcal{R}}^{j|G}$ and $RFI_j^G = 0$.

■

To summarize, we have shown that independence on the dataset and on the model level respectively imply $\text{RFI}_j^G = 0$ and can thereby prove Theorem 1.

Proof of Theorem 1: The result follows from contraposition of Lemma 1 and contraposition of Lemma 2.

■

Theorem 1 shows that nonzero RFI_j^G implies dependencies between sets of variables on the model level as well as on the data level. Which dependencies are relevant for RFI_j^G can be controlled with the conditioning set G. Consequently, the conditioning set G determines the method's implicit definition of relevance. I.e., on the data level, if $X_j \perp\!\!\!\perp (X_{\underline{R}}, Y)|X_G$ holds, RFI_j^G is zero irrespective of any other dependencies that may hold, e.g. with X_G (Lemma 1). Nonzero RFI, a difference in performance on interventional and observational distribution, can only be caused by dependencies that have been destroyed in the interventional distribution, the dependencies with and via X_G are preserved by the replacement $\tilde{X}_j^G$ and can therefore not be responsible for $\text{RFI}_j^G \neq 0$. Similarly, on the model level, $\tilde{X}_j^G \perp\!\!\!\perp \hat{Y}|X_R$ over the interventional distribution $P(X_j|X_G)P(X_G, X_{\underline{R}})$ yields zero RFI (Lemma 2). The behavior of the model outside the domain in which it is evaluated is irrelevant for RFI_j^G. What domain the model is evaluated over depends on the choice of G.
Because we can control RFI's implicit definition of relevance with G, RFI allows more nuanced insights into model and data than PFI or CFI alone. In Theorem 1, we aim to make the implicit definition of relevance explicit. On the data level, nonzero RFI implies the dependence of X_j with the tuple $(Y, X_{\underline{R}})$ given X_G ($X_j \not\perp\!\!\!\perp (Y, X_{\underline{R}})|X_G$). In order to understand the aforementioned dependence, using the graphoid axioms contraction and weak union [22], the equivalent formulation below can be adduced:

$$(X_j \not\perp\!\!\!\perp Y|X_G) \vee (X_j \not\perp\!\!\!\perp X_{\underline{R}}|X_G, Y).$$

At least one of the two conditional dependencies has to hold for nonzero RFI_j^G. The first dependence can be rephrased as: X_j is informative of Y, even if we already know X_G. It is more difficult to make sense of the second dependence. Under dependent features ($X_j \not\perp\!\!\!\perp X_{\underline{R}}|X_G, Y$), the distribution of X_j with $X_{\underline{R}}$ is not preserved under perturbation $\tilde{X}_j^G$. In the interventional distribution $P(\tilde{X}_j^G, X_{\underline{R}})$ observations that are improbable or impossible w.r.t. the observational distribution $P(X_j, X_{\underline{R}})$ can be possible and probable (and vice versa). Consequently, in the interventional distribution the feature distribution differs from the observation feature distribution. Even if $X_j \perp\!\!\!\perp Y|X_G$ holds, the model may perform suboptimally due to this distribution shift and cause RFI_j^G nonzero[2]. If the conditioning set is a superset of R ($G \supseteq R$), such that set of remaining variables $X_{\underline{R}}$ is empty, it holds that ($X_j \perp\!\!\!\perp X_{\underline{R}}|X_G, Y$). Therefore nonzero RFI must be attributed to ($X_j \not\perp\!\!\!\perp Y|X_G$) for $G \supseteq R$.

On the model level, nonzero RFI implies that the model's predictions are conditionally dependent on $\tilde{X}_j^G$ given the remaining features R are fixed. E.g. for a linear model that has coefficient zero for all terms involving X_j, this dependence would not be fulfilled, and RFI_j^G would be zero (Lemma 2). The model is evaluated over the interventional distribution $P(X_j|X_G)P(X_G, X_{\underline{R}}) > 0$, which varies depending on G. If G contains a nearly perfect correlate of X_j, X_j can be reconstructed well. In contrast, if $G = \emptyset$, for every possible x_R the model is evaluated over the whole marginal distribution of X_j. Although choosing a smaller set $G \subset R$ leads to extrapolation under dependent features, it allows more insight into the model's mechanism. For interpretation purposes like safety, this is highly desirable.

In the preceding paragraphs we have highlighted the importance of the conditioning set G for the method's implicit notion of relevance and illustrated the results from Theorem 1. We have argued that the conditioning set controls which potential dependencies can be responsible for nonzero RFI_j^G. The insights lead to a further, interesting application of RFI. By assessing the difference $\Delta RFI_j^{G \to G \cup N} = \mathrm{RFI}_j^G - \mathrm{RFI}_j^{G \cup N}$ when modifying the conditioning set G by adding new elements N, we are able to assess the role of the dependencies with variables in N relative to a baseline G. While for RFI_j^G only dependencies of X_j with and via G are preserved, for $\mathrm{RFI}_j^{G \cup N}$ also dependencies with and via N are maintained. If $\Delta RFI_j^{G \to G \cup N}$ is nonzero, this change has to be due to dependencies involving N, but not G. We substantiate this claim with Theorem 2. In order for $\Delta RFI_j^{G \to G \cup N}$ to be positive, the dependence $X_j \not\perp\!\!\!\perp X_N|X_G$ has to hold.

Theorem 2: If the difference $\Delta RFI_j^{G \to G \cup N} = RFI_j^G$ - $RFI_j^{G \cup N} \neq 0$, then $X_j \not\perp\!\!\!\perp X_N|X_G$.

Proof of Theorem 2: Under independence $X_j \perp\!\!\!\perp X_n|X_G$ it holds that

$$\begin{aligned}
P(\tilde{X}_j^G, Y, X_{\underline{\underline{R}}}, X_G, X_N) &= P(\tilde{X}_j^G|Y, X_{\underline{\underline{R}}}, X_G, X_N)P(Y, X_{\underline{\underline{R}}}, X_G, X_N) \\
&\overset{(\text{def } \tilde{X}_j^G)}{=} P(X_j|X_G)P(Y, X_{\underline{\underline{R}}}, X_G, X_N) \\
&\overset{X_j \perp\!\!\!\perp X_n|X_G}{=} P(X_j|X_G, X_N)P(Y, X_{\underline{\underline{R}}}, X_G, X_N) \\
&\overset{(\text{def } \tilde{X}_j^{G \cup N})}{=} P(\tilde{X}_j^{G \cup N}|X_G, X_N)P(Y, X_{\underline{\underline{R}}}, X_G, X_N) \\
&\overset{(\text{def } \tilde{X}_j^{G \cup N})}{=} P(\tilde{X}_j^{G \cup N}|Y, X_G, X_N, X_{\underline{\underline{R}}})P(Y, X_{\underline{\underline{R}}}, X_G, X_N) \\
&= P(\tilde{X}_j^{G \cup N}, Y, X_{\underline{\underline{R}}}, X_G, X_N)
\end{aligned}$$

The equality $P(\tilde{X}_j^G, Y, X_{\underline{\underline{R}}}, X_G, X_N) = P(\tilde{X}_j^{G \cup N}, Y, X_{\underline{\underline{R}}}, X_G, X_N)$ implies $P(\tilde{X}_j^G, Y, X_R) = (\tilde{X}_j^{G \cup N}, Y, X_R)$. Invoking Proposition 1 it holds that the corresponding risks $\mathcal{R}^{j|G}$ and $\mathcal{R}^{j|G \cup N}$ are equal. As $\mathrm{RFI}_j^G - \mathrm{RFI}_j^{G \cup N} = \mathcal{R}^{j|G} - \mathcal{R}^{j|G \cup N}$ it holds that $X_j \not\perp\!\!\!\perp X_n|X_G \Rightarrow \Delta RFI_j^{G \to G \cup N} = 0$. Contraposition proves Theorem 2.

■

While nonzero RFI_j^G as well as nonzero $\Delta RFI_j^{G \to G \cup N}$ have clear implications, interpreting zero RFI_j^G or zero $\Delta RFI_j^{G \to G \cup N}$ is difficult. For example, we may be tempted to interpret $\mathrm{RFI}_j^G = 0$ as conditional independence in the data. However, the general principle that absence of evidence is no evidence for absence also applies in the context of RFI. A dependence in the data may not be captured by the model when it has a poor fit and does not rely on the respective variable. Similarly, although f may be optimal, a dependence in higher moments may simply not be modeled by f or captured by the loss L. As all aforementioned causes of nonzero RFI are potentially sufficient, but not necessary, it is unclear which of the causes nonzero RFI can be attributed to. Furthermore, the related problem of conditional independence testing is provably hard [24].

The theoretical insights that we derive in this Section (Theorem 1 and 2) are applied and illustrated in a simulation study in Section VI.

V. Estimation and Testing

Estimating and sampling from the conditional distribution is in general difficult, especially in high-dimensional continuous settings. Various approaches for replacing X_j with samples from its conditional distribution exist, e.g., knockoff approaches [2], [7], [23], imputation and weighting [12] or permutation within decision tree leaves [18]. We used Model-X knockoffs [7] in this work, but note that the RFI approach is agnostic to its algorithmic implementation.

[2] Let e.g. X_1, X_2 be perfectly correlated and independent of Y. Then adding $X_1 - X_2$ does not alter its prediction performance, unless the dependence between the variables is broken. Also see [14] for a discussion in PFI.

Using (standard) empirical risk estimates, our RFI estimate is

$$\hat{\mathrm{RFI}}_j^G = \frac{1}{n}\sum_{i=1}^{n} L\left(y^{(i)}, f(\tilde{x}_j^{(i)}, x_R^{(i)})\right) - \frac{1}{n}\sum_{i=1}^{n} L\left(y^{(i)}, f(x_j^{(i)}, x_R^{(i)})\right)$$

where $\tilde{x}_j^{(i)}$ is a sample from $\tilde{X}_j^G$. We can then test for nonzero RFI_j^G using procedures for conditional independence tests, e.g., [30], thereby quantifying the uncertainty coming from empirical risk minimization. Because of the central limit theorem, the empirical risk converges (in probability) to a Gaussian distribution with increasing number of observations. Therefore, one-sided, paired t-tests can be used to infer tests and confidence intervals [30]. The test procedures proposed in [30] are agnostic to the conditioning set for the perturbation $\tilde{X}_j^G$. For smaller samples, the Exact Test by Fisher may be used.

The t-test and Fisher Exact Test ignore uncertainty and bias of the estimation procedures, i.e. the ML model and the knockoff-sampler are treated as "fixed". E.g. misspecified, suboptimal models may not capture dependencies. Or dependencies are in higher moments that are not captured by the loss. Consequently, without further assumptions, the framework does not provide a test for conditional independence in the dataset.

The popular testing procedures for knockoffs proposed by [7] provide FDR over all features, but does not test the significance of the importance of individual features.

VI. Simulation Studies

In the following, we demonstrate the usefulness of RFI on two simulation studies. In the first example, we use RFI to expose indirect influence of variables that are not computationally used by the model. In the second example, we assess feature importance relative to a confounder that was unavailable at training time. In both examples, we represent the underlying data generating mechanism, that gives rise to the dependencies in the data, with a causal directed acyclic graph (DAG). The code for the examples is available online[3].

A. Indirect Influence

A prominent application of interpretable machine learning is auditing models regarding its reliance on protected attributes A like age or sex. A reliance on the respective attributes may result in unfair discrimination and requires further inspection. With approaches like fairness through unawareness [3], the model does not rely on protected attributes directly. However, by implicitly reconstructing the sensitive attributes using seemingly harmless correlates, the model can indirectly make use of the protected attribute resulting in potentially harmful, unfair discrimination [3]. PFI and CFI cannot expose such indirect influence. As Lemma 2 proves, RFI_A^G is zero for a model that does not (directly) use the feature of interest A for the prediction for any conditioning set G. Furthermore, from PFI and CFI alone, we cannot infer whether the importance of a variable can be attributed to its dependence with an indirect influence. Using RFI_j^G with $G = A$ we preserve the influence of A on the prediction and can thereby restrict the attribution of importance to contributions stemming from dependencies not involving A (Theorem 1, Lemma 1). The difference to $\Delta RFI_j^{G \rightarrow G \cup N}$ with $G = \emptyset$ and $N = A$ exposes the indirect influence.

Not every indirect influence from a sensitive attribute is considered undesirable. Certain correlates of A may indeed be valid criteria for a decision (e.g. [4]). Importance stemming from dependencies with A via such resolving variables Z would be considered acceptable. We can assess the indirect influence beyond contributions stemming from dependence via Z by comparing to a baseline $G = Z$. In this baseline, contributions via Z are preserved and therefore irrelevant for RFI. Consequently, when setting $N = A$, the difference $\Delta RFI_j^{G \rightarrow G \cup N}$ only quantifies indirect influence that is not resolved by Z.

We demonstrate the usefulness of RFI to expose indirect influence in a simulation study. The dataset is a sample drawn from the distribution induced by a structural causal model (SCM) depicted in Figure 2. All relationships are additive linear with coefficients 1 and Gaussian noise terms ($\sigma_1 = \sigma_2 = \sigma_4 = 1$, $\sigma_3 = 0.3$ and $\sigma_y = 0.5$). An ordinary least squares linear regression model was fit to predict Y from $X_1, \ldots, X4$ (MSE = 0.25, $f(x_1, x_2, x_3, x_4) = 0.00x_1 - 0.01x_2 + 1.01x_3 + 1.00x_4$). We trained model-X knockoffs [7] on the training data and evaluated RFI on test data. Sample size is 10^5 with 10% test data.

In order to quantify the direct influence of the features we compute PFI. As we can see in Figure 3, X_1 and X_2 are considered irrelevant. In order to expose their indirect influence, we additionally compute RFI with respect to $G = \{X_1\}$ and $G = \{X_2\}$ respectively. For both variables we observe a drop in importance of X_3 and X_4. Consequently both X_1 and X_2 have an indirect influence on the target (Theorem 2).

Furthermore we are interested in whether the indirect influence of X_1 can be resolved by X_2. We therefore compute $\mathrm{RFI}_j^{G \cup N}$ with $G = \{X_2\}$ and $N = \{X_1\}$. We see that for X_3 no change in importance can be observed. This is due to the independence $X_1 \perp\!\!\!\perp X_3 | X_2$[4] (Theorem 2). The indirect influence is resolved. However, for X_4 the importance decreases further and is therefore not resolved by X_2. This is in alignment with the dependence $X_1 \not\perp\!\!\!\perp X_4 | X_2$ implied by the graph (Figure 2).

[3] Link to Code: `https://github.com/gcskoenig/icpr2020-rfi`

[4] As faithfulness and causal markov condition hold, d-separation in the graph and (conditional) independence coincide [21]. We can therefore read the independence structures off Figures 2 and 4.

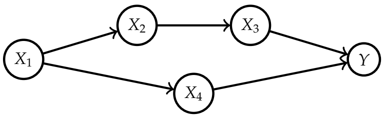


Fig. 2. Variable X_1 influences Y both via the chain $X_2 \rightarrow X_3$ and via X_4. X_1 may be some undesired influence, and X_2 a variable resolving the undesired influence. We find that the prediction can nevertheless be influenced via X_4 by comparing $RFI_4^{X_2}$ with $RFI_4^{X_2,X_1}$ (Figure 3). All relationships are additive linear Gaussian with all coefficients being equal to 1 and $\sigma_1 = \sigma_2 = \sigma_4 = 1$, $\sigma_3 = 0.3$ and $\sigma_y = 0.5$.

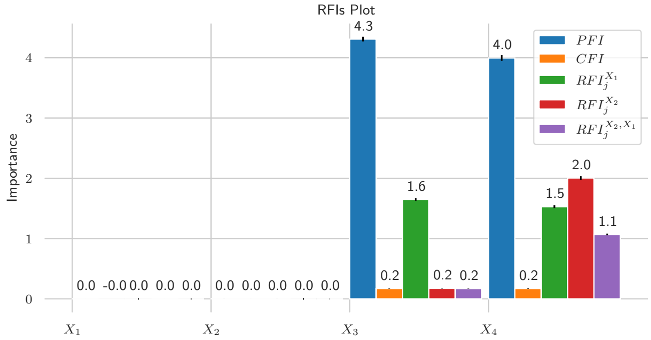


Fig. 3. RFI's for a linear regression model fitted on the dataset illustrated in Figure 2. Feature importance values are averaged over 30 runs and rounded. Feature importance values are averaged over 30 runs and rounded. We evaluated significance using a t-test for the first run. All positive features were significant at $\alpha = 0.01$, whereas for all zero RFI values the null could not be rejected. For X_1 and X_2 all RFIs are zero, whereas for X_3 and X_4 RFIs are positive. We see that X_1 and X_2 both have an indirect influence on X_3 and X_4, but that X_2 can resolve the influence of X_1 on X_3.

B. *Variables Outside Training Set*

When designing a model f, a practitioner may have decided to exclude a variable from the feature set, e.g., because it was then considered irrelevant, it belongs to a different modality or would have required further preprocessing. Furthermore, when auditing a machine learning model f, variables that have not been available for the training of the model may be accessible.

In this example, we demonstrate that variables outside the training set can be included in the conditioning set for RFI. Consequently, importance of the features relative to variables outside the training set and the indirect influence of such variables can be assessed. More specifically, we simulate a hypothetical situation where the influence of a previously unknown confounder C shall be evaluated. This variable C is available for the model audit. In particular, we wonder whether the features X_1, X_2 and X_3 are only or partly important due to a dependence via C.

The dataset was sampled from a structural causal model (SCM) depicted in Figure 4. Assuming faithfulness and the causal Markov condition, this DAG implies the following (conditional) (in-)dependencies: X_1 is independent of C, X_3 is independent of Y conditional on C, and X_2 is dependent on Y. Note that the dependence between X_2 and Y is due to the common cause C as well as due to a direct effect of X_2 on Y. All relationships are additive linear with coefficients 1 and additive Gaussian noise ($\sigma_1 = \sigma_2 = \sigma_C = 1.0$ and $\sigma_3 = \sigma_Y = 0.5$). We fit an ordinary least squares linear regression model on X_1, X_2 and X_3 to predict Y (MSE = 0.40, $f(x_1, x_2, x_3) = 1.0x_1 + 1.17x_2 + 0.67x_3$). C was not available for model training. We trained Model-X knockoffs [7] on training data and sampled from $\tilde{X}_j^G$ on test data. Sample size is 10^5 with 10% test data.

When computing RFI_j^C ($G = \{C\}$) for each variable, the different relationships with C become apparent. The respective results are depicted in Figure 5. For X_1 the feature importance relative to C remains unchanged as the variables are pairwise independent (Theorem 2). For X_3, that is only dependent with Y via C, it completely vanishes (Lemma 1). For X_2 the feature importance decreases but remains nonzero, as X_2 is dependent with Y directly and via C.

Consequently, using RFI, we can (1) identify variables that are important due to a variable unavailable at training time and (2) distinguish between variables that only depend on Y via C from those that do not. With PFI ($G = \emptyset$) or CFI ($G = R$) such a distinction is in general not possible.

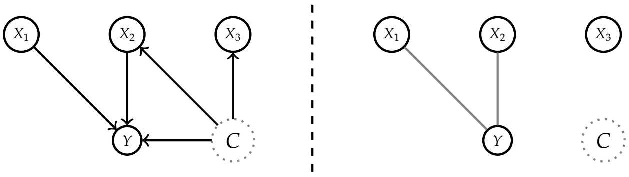


Fig. 4. *Left:* We see the causal graph $\mathcal{G}$ corresponding to the Structural Causal Model that was used to generate the dataset used in Figure 5. All relationships are additive linear Gaussian with all coefficients equal to 1 and $\sigma_1 = \sigma_2 = \sigma_C = 1.0$ and $\sigma_3 = \sigma_Y = 0.5$. *Right:* Pairwise dependencies after conditioning on C.

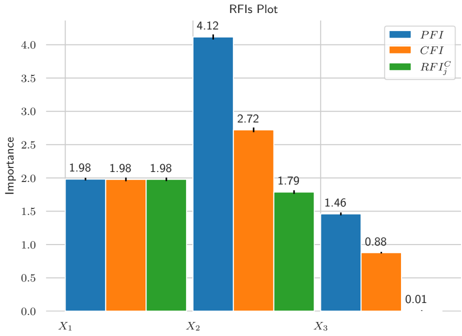


Fig. 5. Feature Importance results corresponding to the dataset depicted in Figure 4. We averaged RFI over 30 runs. RFI for X_1 is unaffected by changes in G, for X_2 RFI drops with C is added to G. For X_3 RFI vanishes relative to C. For all except for $RFI_{X_3}^C$ the null can be rejected at $\alpha = 0.01$ in the first run.

VII. Discussion

We proposed relative feature importance (RFI), a general conditional feature importance framework which allows to condition on arbitrary sets of other features, including features outside the training set. We underpin the method with theoretical results allowing insight into both model and underlying dataset. In a simulation study, the usefulness of the method for the exposure of indirect influence is demonstrated.

Relative feature importance requires sampling from (unknown) conditional distributions. For continuous variables and in high-dimensional settings this task is challenging and an open area of research [7], [23]. Uncertainty stemming from inaccurate sampling may affect the interpretation. The quality of insight into the underlying dataset strongly depends on the training and evaluation of the model. Dependencies in higher moments are usually not modeled and not captured by standard loss functions and can therefore not be detected. Especially the interpretation of zero RFI requires careful assessment of the model specification. Further research is needed to assess necessary assumptions for the interpretation of RFI. These challenges are not unique to RFI, but apply more generally in the field of interpretable machine learning [20].

References

[1] Philip Adler, Casey Falk, Sorelle A. Friedler, Tionney Nix, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. Auditing black-box models for indirect influence. Knowledge and Information Systems, 54(1):95–122, 2018. arXiv: 1602.07043.

[2] Rina Foygel Barber, Emmanuel J Cands, and others. Controlling the false discovery rate via knockoffs. The Annals of Statistics, 43(5):2055–2085, 2015. Publisher: Institute of Mathematical Statistics.

[3] Solon Barocas, Moritz Hardt, and Arvind Narayanan. Fairness and Machine Learning. fairmlbook.org, 2019. http://www.fairmlbook.org.

[4] Vence L Bonham, Shawneequa L Callier, and Charmaine D Royal. Will precision medicine move us beyond race? The New England journal of medicine, 374(21):2003, 2016.

[5] Leo Breiman. Random forests. Machine Learning, pages 1–122, 2001.

[6] David V Budescu. Dominance analysis: a new approach to the problem of relative importance of predictors in multiple regression. Psychological bulletin, 114(3):542, 1993.

[7] Emmanuel Cands, Yingying Fan, Lucas Janson, and Jinchi Lv. Panning for gold: model-X knockoffs for high dimensional controlled variable selection. Journal of the Royal Statistical Society. Series B: Statistical Methodology, 80(3):551–577, 2018. arXiv: 1610.02351.

[8] Ian Covert, Scott Lundberg, and Su-In Lee. Understanding Global Feature Contributions Through Additive Importance Measures. arXiv preprint arXiv:2004.00668, 2020.

[9] Jeffrey (Reuters) Dastin. Amazon scraps secret AI recruiting tool that showed bias against women. Reuters, 2018.

[10] Lydia de la Torre. A Guide to the California Consumer Privacy Act of 2018. SSRN Electronic Journal, pages 1–17, 2018.

[11] Julia Dressel and Hany Farid. The accuracy, fairness, and limits of predicting recidivism. Science Advances, 4(1):1–6, 2018.

[12] Aaron Fisher, Cynthia Rudin, and Francesca Dominici. All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research, 20(177):1–81, 2019.

[13] Ulrike Grmping. Variable importance assessment in regression: linear regression versus random forest. The American Statistician, 63(4):308–319, 2009. Publisher: Taylor & Francis.

[14] Giles Hooker and Lucas Mentch. Please Stop Permuting Features: An Explanation and Alternatives. arXiv preprint arXiv:1905.03151v, pages 1–15, 2019. arXiv: 1905.03151v1.

[15] Jing Lei, Max GSell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. Journal of the American Statistical Association, 113(523):1094–1111, 2018.

[16] Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. Applied Stochastic Models in Business and Industry, 17(4):319–330, 2001. Lipovetsky2001.

[17] Scott M. Lundberg and Su In Lee. A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 2017-Decem(Section 2):4766–4775, 2017. arXiv: 1705.07874.

[18] Christoph Molnar, Gunnar König, Bernd Bischl, and Giuseppe Casalicchio. Model-agnostic feature importance and effects with dependent features–a conditional subgroup approach. arXiv preprint arXiv:2006.04628, 2020.

[19] Christoph Molnar, Gunnar König, Bernd Bischl, and Giuseppe Casalicchio. Model-agnostic feature importance and effects with dependent features-a conditional subgroup approach. arXiv preprint arXiv:2006.04628, 2020.

[20] Christoph Molnar, Gunnar König, Julia Herbinger, Timo Freiesleben, Susanne Dandl, Christian A. Scholbeck, Giuseppe Casalicchio, Moritz Grosse-Wentrup, and Bernd Bischl. Pitfalls to avoid when interpreting machine learning models. arXiv preprint arXiv:2007.04131, 2020.

[21] Judea Pearl. Causality. Cambridge university press, 2009.

[22] Judea Pearl and Azaria Paz. Graphoids: A graph-based logic for reasoning about relevance relations. University of California (Los Angeles). Computer Science Department, 1985.

[23] Yaniv Romano, Matteo Sesia, and Emmanuel Cands. Deep knockoffs. Journal of the American Statistical Association, pages 1–12, 2019. Publisher: Taylor & Francis.

[24] Rajen D Shah and Jonas Peters. The hardness of conditional independence testing and the generalised covariance measure. arXiv preprint arXiv:1804.07203, 2018.

[25] Carolin Strobl, Anne Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. Conditional variable importance for random forests. BMC Bioinformatics, 9:1–11, 2008.

[26] Laura ToloÅi and Thomas Lengauer. Classification with correlated features: unreliability of feature ranking and solutions. Bioinformatics, 27(14):1986–1994, 2011. Publisher: Oxford University Press.

[27] Eric J Topol. High performance medicine: the convergence of human and artificial intelligence. Nature Medicine, 25(January), 2019. Publisher: Springer US.

[28] Eugene Tuv, Alexander Borisov, George Runger, and Kari Torkkola. Feature selection with ensembles, artificial variables, and redundancy elimination. Journal of Machine Learning Research, 10(Jul):1341–1366, 2009.

[29] Paul Voigt and Axel dem Bussche. The eu general data protection regulation (gdpr). A Practical Guide, 1st Ed., Cham: Springer International Publishing, 2017. Publisher: Springer.

[30] David S. Watson and Marvin N. Wright. Testing Conditional Independence in Supervised Learning Algorithms. arXiv preprint arXiv:1901.09917, 2019. arXiv: 1901.09917.

[31] Pengfei Wei, Zhenzhou Lu, and Jingwen Song. Variable importance analysis: a comprehensive review. Reliability Engineering & System Safety, 142:399–432, 2015. Publisher: Elsevier.

[32] Yufei Xia, Chuanzhe Liu, Yu Ying Li, and Nana Liu. A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. Expert Systems with Applications, 78:225–241, 2017. Publisher: Elsevier Ltd.

[33] Erik Åtrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. Knowledge and information systems, 41(3):647–665, 2014. Publisher: Springer.

2.5 Paper V: General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models

Molnar, Christoph, Gunnar König, Julia Herbinger, Timo Freiesleben, Susanne Dandl, Christian Scholbeck, Giuseppe Casalicchio, Moritz Grosse-Wentrup and Bernd Bischl (2022). **General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models.** *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers.* Cham: Springer International Publishing, 2020.

Gunnar König contributed to the paper as a co-author with significant contributions. Gunnar König suggested the framing of the paper and wrote large parts of Sections 2, 5, and 10. Christoph Molnar initiated and coordinated the project. The co-authors mainly wrote the remaining sections. All authors added input to the chapters in which they were not involved as authors and proofread and revised the paper.

General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models

Christoph Molnar[1,7]( ), Gunnar König[1,4], Julia Herbinger[1], Timo Freiesleben[2,3], Susanne Dandl[1], Christian A. Scholbeck[1], Giuseppe Casalicchio[1], Moritz Grosse-Wentrup[4,5,6], and Bernd Bischl[1]

[1] Department of Statistics, LMU Munich, Munich, Germany
christoph.molnar.ai@gmail.com
[2] Munich Center for Mathematical Philosophy, LMU Munich, Munich, Germany
[3] Graduate School of Systemic Neurosciences, LMU Munich, Munich, Germany
[4] Research Group Neuroinformatics, Faculty for Computer Science, University of Vienna, Vienna, Austria
[5] Research Platform Data Science @ Uni Vienna, Vienna, Austria
[6] Vienna Cognitive Science Hub, Vienna, Austria
[7] Leibniz Institute for Prevention Research and Epidemiology - BIPS GmbH, Bremen, Germany

Abstract. An increasing number of model-agnostic interpretation techniques for machine learning (ML) models such as partial dependence plots (PDP), permutation feature importance (PFI) and Shapley values provide insightful model interpretations, but can lead to wrong conclusions if applied incorrectly. We highlight many general pitfalls of ML model interpretation, such as using interpretation techniques in the wrong context, interpreting models that do not generalize well, ignoring feature dependencies, interactions, uncertainty estimates and issues in high-dimensional settings, or making unjustified causal interpretations, and illustrate them with examples. We focus on pitfalls for global methods that describe the average model behavior, but many pitfalls also apply to local methods that explain individual predictions. Our paper addresses ML practitioners by raising awareness of pitfalls and identifying solutions for correct model interpretation, but also addresses ML researchers by discussing open issues for further research.

Keywords: Interpretable machine learning · Explainable AI

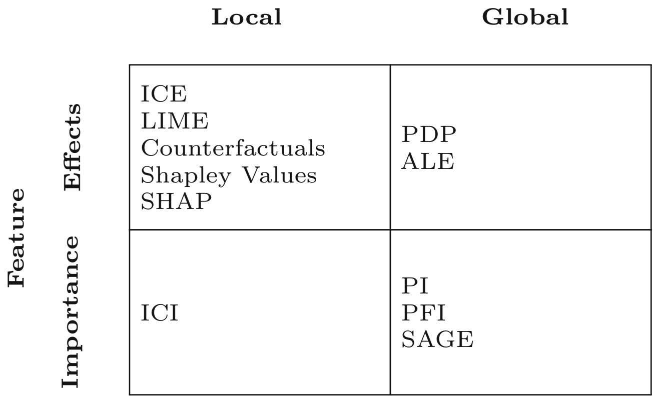
This work is funded by the Bavarian State Ministry of Science and the Arts (coordinated by the Bavarian Research Institute for Digital Transformation (bidt)), by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A, by the German Research Foundation (DFG), Emmy Noether Grant 437611051, and by the Graduate School of Systemic Neurosciences (GSN) Munich. The authors of this work take full responsibilities for its content.

A. Holzinger et al. (Eds.): xxAI 2020, LNAI 13200, pp. 39–68, 2022.
https://doi.org/10.1007/978-3-031-04083-2_4

1 Introduction

In recent years, both industry and academia have increasingly shifted away from parametric models, such as generalized linear models, and towards non-parametric and non-linear machine learning (ML) models such as random forests, gradient boosting, or neural networks. The major driving force behind this development has been a considerable outperformance of ML over traditional models on many prediction tasks [32]. In part, this is because most ML models handle interactions and non-linear effects automatically. While classical statistical models – such as generalized additive models (GAMs) – also support the inclusion of interactions and non-linear effects, they come with the increased cost of having to (manually) specify and evaluate these modeling options. The benefits of many ML models are partly offset by their lack of interpretability, which is of major importance in many applications. For certain model classes (e.g. linear models), feature effects or importance scores can be directly inferred from the learned parameters and the model structure. In contrast, it is more difficult to extract such information from complex non-linear ML models that, for instance, do not have intelligible parameters and are hence often considered black boxes. However, model-agnostic interpretation methods allow us to harness the predictive power of ML models while gaining insights into the black-box model. These interpretation methods are already applied in many different fields. Applications of interpretable machine learning (IML) include understanding pre-evacuation decision-making [124] with partial dependence plots [36], inferring behavior from smartphone usage [105,106] with the help of permutation feature importance [107] and accumulated local effect plots [3], or understanding the relation between critical illness and health records [70] using Shapley additive explanations (SHAP) [78]. Given the widespread application of interpretable machine learning, it is crucial to highlight potential pitfalls, that, in the worst case, can produce incorrect conclusions.

This paper focuses on pitfalls for model-agnostic IML methods, i.e. methods that can be applied to any predictive model. Model-specific methods, in contrast, are tied to a certain model class (e.g. saliency maps [57] for gradient-based models, such as neural networks), and are mainly considered out-of-scope for this work. We focus on pitfalls for global interpretation methods, which describe the expected behavior of the entire model with respect to the whole data distribution. However, many of the pitfalls also apply to local explanation methods, which explain individual predictions or classifications. Global methods include the partial dependence plot (PDP) [36], partial importance (PI) [19], accumulated local affects (ALE) [3], or the permutation feature importance (PFI) [12,19,33]. Local methods include the individual conditional expectation (ICE) curves [38], individual conditional importance (ICI) [19], local interpretable model-agnostic explanations (LIME) [94], Shapley values [108] and SHapley Additive exPlanations (SHAP) [77,78] or counterfactual explanations [26,115]. Furthermore, we distinguish between feature effect and feature importance methods. A feature effect indicates the direction and magnitude of a change in predicted outcome due to changes in feature values. Effect methods include



Feature	Local	Global
Effects	ICE LIME Counterfactuals Shapley Values SHAP	PDP ALE
Importance	ICI	PI PFI SAGE

Fig. 1. Selection of popular model-agnostic interpretation techniques, classified as local or global, and as effect or importance methods.

Shapley values, SHAP, LIME, ICE, PDP, or ALE. Feature importance methods quantify the contribution of a feature to the model performance (e.g. via a loss function) or to the variance of the prediction function. Importance methods include the PFI, ICI, PI, or SAGE. See Fig. 1 for a visual summary.

The interpretation of ML models can have subtle pitfalls. Since many of the interpretation methods work by similar principles of manipulating data and "probing" the model [100], they also share many pitfalls. The sources of these pitfalls can be broadly divided into three categories: (1) application of an unsuitable ML model which does not reflect the underlying data generating process very well, (2) inherent limitations of the applied IML method, and (3) wrong application of an IML method. Typical pitfalls for (1) are bad model generalization or the unnecessary use of complex ML models. Applying an IML method in a wrong way (3) often results from the users' lack of knowledge of the inherent limitations of the chosen IML method (2). For example, if feature dependencies and interactions are present, potential extrapolations might lead to misleading interpretations for perturbation-based IML methods (inherent limitation). In such cases, methods like PFI might be a wrong choice to quantify feature importance.

Table 1. Categorization of the pitfalls by source.

Sources of pitfall	Sections
Unsuitable ML model	3, 4
Limitation of IML method	5.1, 6.1, 6.2, 9.1, 9.2
Wrong application of IML method	2, 5.2, 5.3, 7, 8, 9.3, 10

Contributions: We uncover and review general pitfalls of model-agnostic interpretation techniques. The categorization of these pitfalls into different sources is provided in Table 1. Each section describes and illustrates a pitfall, reviews possible solutions for practitioners to circumvent the pitfall, and discusses open issues that require further research. The pitfalls are accompanied by illustrative examples for which the code can be found in this repository: https://github.com/compstat-lmu/code_pitfalls_iml.git. In addition to reproducing our examples, we invite readers to use this code as a starting point for their own experiments and explorations.

Related Work: Rudin et al. [96] present principles for interpretability and discuss challenges for model interpretation with a focus on inherently interpretable models. Das et al. [27] survey methods for explainable AI and discuss challenges with a focus on saliency maps for neural networks. A general warning about using and explaining ML models for high stakes decisions has been brought forward by Rudin [95], in which the author argues against model-agnostic techniques in favor of inherently interpretable models. Krishnan [64] criticizes the general conceptual foundation of interpretability, but does not dispute the usefulness of available methods. Likewise, Lipton [73] criticizes interpretable ML for its lack of causal conclusions, trust, and insights, but the author does not discuss any pitfalls in detail. Specific pitfalls due to dependent features are discussed by Hooker [54] for PDPs and functional ANOVA as well as by Hooker and Mentch [55] for feature importance computations. Hall [47] discusses recommendations for the application of particular interpretation methods but does not address general pitfalls.

2 Assuming One-Fits-All Interpretability

Pitfall: Assuming that a single IML method fits in all interpretation contexts can lead to dangerous misinterpretation. IML methods condense the complexity of ML models into human-intelligible descriptions that only provide insight into specific aspects of the model and data. The vast number of interpretation methods make it difficult for practitioners to choose an interpretation method that can answer their question. Due to the wide range of goals that are pursued under the umbrella term “interpretability”, the methods differ in which aspects of the model and data they describe.

For example, there are several ways to quantify or rank the features according to their relevance. The relevance measured by PFI can be very different from the relevance measured by the SHAP importance. If a practitioner aims to gain insight into the relevance of a feature regarding the model’s generalization error, a loss-based method (on unseen test data) such as PFI should be used. If we aim to expose which features the model relies on for its prediction or classification – irrespective of whether they aid the model’s generalization performance – PFI on test data is misleading. In such scenarios, one should quantify the relevance of a feature regarding the model’s prediction (and not the model’s generalization error) using methods like the SHAP importance [76].

We illustrate the difference in Fig. 2. We simulated a data-generating process where the target is completely independent of all features. Hence, the features are just noise and should not contribute to the model’s generalization error. Consequently, the features are not considered relevant by PFI on test data.

However, the model mechanistically relies on a number of spuriously correlated features. This reliance is exposed by marginal global SHAP importance.

As the example demonstrates, it would be misleading to view the PFI computed on test data or global SHAP as one-fits-all feature importance techniques. Like any IML method, they can only provide insight into certain aspects of model and data.

Many pitfalls in this paper arise from situations where an IML method that was designed for one purpose is applied in an unsuitable context. For example, extrapolation (Sect. 5.1) can be problematic when we aim to study how the model behaves under realistic data but simultaneously can be the correct choice if we want to study the sensitivity to a feature outside the data distribution.

For some IML techniques – especially local methods – even the same method can provide very different explanations, depending on the choice of hyperparameters: For counterfactuals, explanation goals are encoded in their optimization metrics [26,34] such as sparsity and data faithfulness; The scope and meaning of LIME explanations depend on the kernel width and the notion of complexity [8,37].

Solution: The suitability of an IML method cannot be evaluated with respect to one-fits-all interpretability but must be motivated and assessed with respect to well-defined interpretation goals. Similarly, practitioners must tailor the choice of the IML method and its respective hyperparameters to the interpretation context. This implies that these goals need to be clearly stated in a detailed manner *before* any analysis – which is still often not the case.

Open Issues: Since IML methods themselves are subject to interpretation, practitioners must be informed about which conclusions can or cannot be drawn given different choices of IML technique. In general, there are three aspects to be considered: (a) an intuitively understandable and plausible algorithmic construction of the IML method to achieve an explanation; (b) a clear mathematical axiomatization of interpretation goals and properties, which are linked by proofs and theoretical considerations to IML methods, and properties of models and data characteristics; (c) a practical translation for practitioners of the axioms from (b) in terms of what an IML method provides and what not, ideally with implementable guidelines and diagnostic checks for violated assumptions to guarantee correct interpretations. While (a) is nearly always given for any published method, much work remains for (b) and (c).

3 Bad Model Generalization

Pitfall: Under- or overfitting models can result in misleading interpretations with respect to the true feature effects and importance scores, as the model does not match the underlying data-generating process well [39]. Formally, most IML methods are designed to interpret the model instead of drawing inferences about

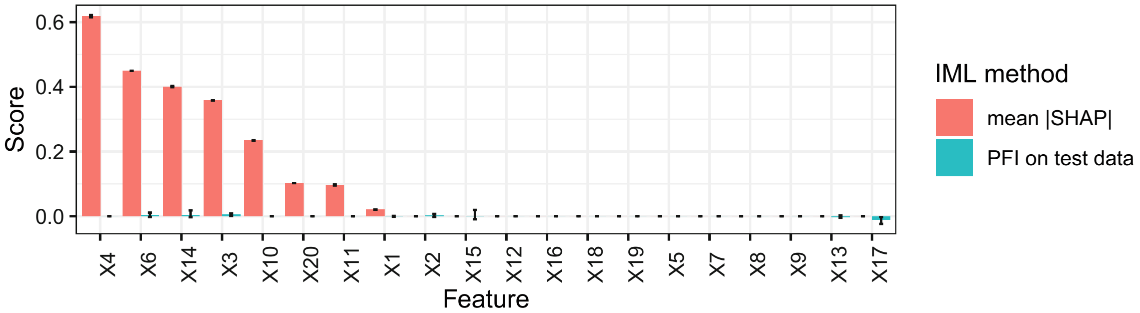


Fig. 2. Assuming one-fits-all interpretability. A default `xgboost` regression model that minimizes the mean squared error (MSE) was fitted on 20 independently and uniformly distributed features to predict another independent, uniformly sampled target. In this setting, predicting the (unconditional) mean $\mathbb{E}[Y]$ in a constant model is optimal. The learner overfits due to a small training data size. Mean marginal SHAP (red, error bars indicate 0.05 and 0.95 quantiles) exposes all mechanistically used features. In contrast, PFI on test data (blue, error bars indicate 0.05 and 0.95 quantiles) considers all features to be irrelevant, since no feature contributes to the generalization performance.

the data-generating process. In practice, however, the latter is often the goal of the analysis, and then an interpretation can only be as good as its underlying model. If a model approximates the data-generating process well enough, its interpretation should reveal insights into the underlying process.

Solution: In-sample evaluation (i.e. on training data) should not be used to assess the performance of ML models due to the risk of overfitting on the training data, which will lead to overly optimistic performance estimates. We must resort to out-of-sample validation based on resampling procedures such as hold-out for larger datasets or cross-validation, or even repeated cross-validation for small sample size scenarios. These resampling procedures are readily available in software [67,89], and well-studied in theory as well as practice [4,11,104], although rigorous analysis of cross-validation is still considered an open problem [103]. Nested resampling is necessary, when computational model selection and hyperparameter tuning are involved [10]. This is important, as the Bayes error for most practical situations is unknown, and we cannot make absolute statements about whether a model already optimally fits the data.

Figure 3 shows the mean squared errors for a simulated example on both training and test data for a support vector machine (SVM), a random forest, and a linear model. Additionally, PDPs for all models are displayed, which show to what extent each model's effect estimates deviate from the ground truth. The linear model is unable to represent the non-linear relationship, which is reflected in a high error on both test and training data and the linear PDPs. In contrast, the random forest has a low training error but a much higher test error, which indicates overfitting. Also, the PDPs for the random forest display overfitting behavior, as the curves are quite noisy, especially at the lower and upper value

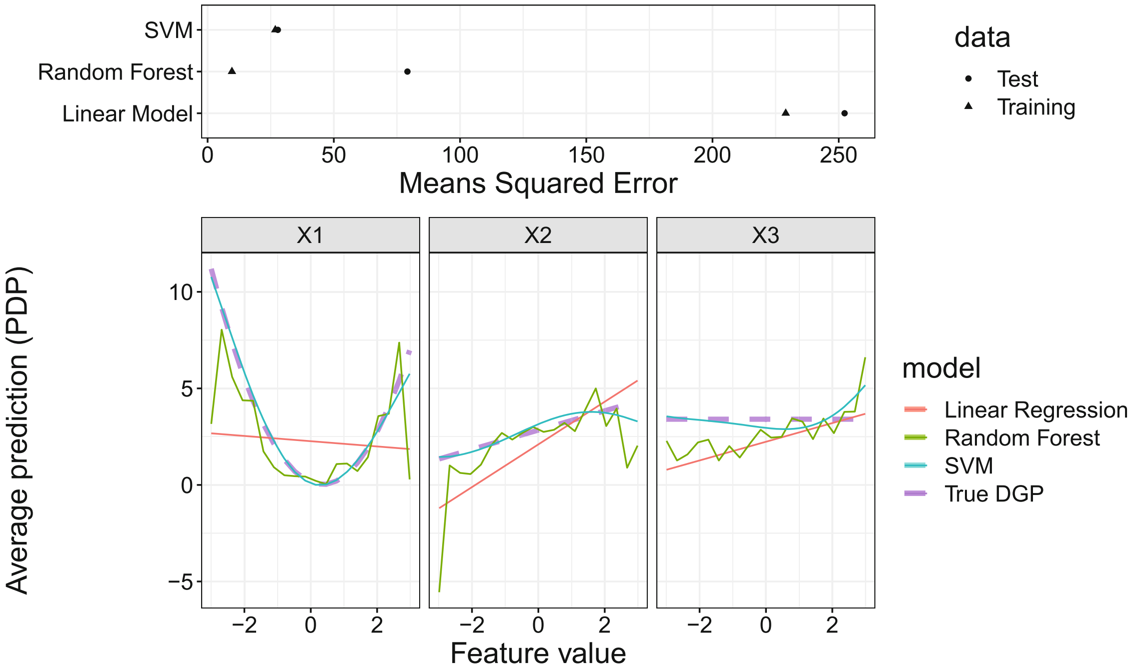


Fig. 3. Bad model generalization. **Top:** Performance estimates on training and test data for a linear regression model (underfitting), a random forest (overfitting) and a support vector machine with radial basis kernel (good fit). The three features are drawn from a uniform distribution, and the target was generated as $Y = X_1^2 + X_2 - 5X_1X_2 + \epsilon$, with $\epsilon \sim N(0,5)$.**Bottom:** PDPs for the data-generating process (DGP) – which is the ground truth – and for the three models.

ranges of each feature. The SVM with both low training and test error comes closest to the true PDPs.

4 Unnecessary Use of Complex Models

Pitfall: A common mistake is to use an opaque, complex ML model when an interpretable model would have been sufficient, i.e. when the performance of interpretable models is only negligibly worse – or maybe the same or even better – than that of the ML model. Although model-agnostic methods can shed light on the behavior of complex ML models, inherently interpretable models still offer a higher degree of transparency [95] and considering them increases the chance of discovering the true data-generating function [23]. What constitutes an interpretable model is highly dependent on the situation and target audience, as even a linear model might be difficult to interpret when many features and interactions are involved.

It is commonly believed that complex ML models always outperform more interpretable models in terms of accuracy and should thus be preferred. However, there are several examples where interpretable models have proven to be serious competitors: More than 15 years ago, Hand [49] demonstrated that simple models often achieve more than 90% of the predictive power of potentially highly complex models across the UCI benchmark data repository and concluded that such models often should be preferred due to their inherent interpretability; Makridakis et al. [79] systematically compared various ML models (including long-short-term-memory models and multi-layer neural networks) to statistical models (e.g. damped exponential smoothing and the Theta method) in time series forecasting tasks and found that the latter consistently show greater predictive accuracy; Kuhle et al. [65] found that random forests, gradient boosting and neural networks did not outperform logistic regression in predicting fetal growth abnormalities; Similarly, Wu et al. [120] have shown that a logistic regression model performs as well as AdaBoost and even better than an SVM in predicting heart disease from electronic health record data; Baesens et al. [7] showed that simple interpretable classifiers perform competitively for credit scoring, and in an update to the study the authors note that "the complexity and/or recency of a classifier are misleading indicators of its prediction performance" [71].

Solution: We recommend starting with simple, interpretable models such as linear regression models and decision trees. Generalized additive models (GAM) [50] can serve as a gradual transition between simple linear models and more complex machine learning models. GAMs have the desirable property that they can additively model smooth, non-linear effects and provide PDPs out-of-the-box, but without the potential pitfall of masking interactions (see Sect. 6). The additive model structure of a GAM is specified before fitting the model so that only the pre-specified feature or interaction effects are estimated. Interactions between features can be added manually or algorithmically (e.g. via a forward greedy search) [18]. GAMs can be fitted with component-wise boosting [99]. The boosting approach allows to smoothly increase model complexity, from sparse linear models to more complex GAMs with non-linear effects and interactions. This smooth transition provides insight into the tradeoffs between model simplicity and performance gains. Furthermore, component-wise boosting has an in-built feature selection mechanism as the model is build incrementally, which is especially useful in high-dimensional settings (see Sect. 9.1). The predictive performance of models of different complexity should be carefully measured and compared. Complex models should only be favored if the additional performance gain is both significant and relevant – a judgment call that the practitioner must ultimately make. Starting with simple models is considered best practice in data science, independent of the question of interpretability [23]. The comparison of predictive performance between model classes of different complexity can add further insights for interpretation.

Open Issues: Measures of model complexity allow quantifying the trade-off between complexity and performance and to automatically optimize for multiple objectives beyond performance. Some steps have been made towards quantifying model complexity, such as using functional decomposition and quantifying the complexity of the components [82] or measuring the stability of predictions [92]. However, further research is required, as there is no single perfect definition of interpretability, but rather multiple depending on the context [30,95].

5 Ignoring Feature Dependence

5.1 Interpretation with Extrapolation

Pitfall: When features are dependent, perturbation-based IML methods such as PFI, PDP, LIME, and Shapley values extrapolate in areas where the model was trained with little or no training data, which can cause misleading interpretations [55]. This is especially true if the ML model relies on feature interactions [45] – which is often the case. Perturbations produce artificial data points that are used for model predictions, which in turn are aggregated to produce global or local interpretations [100]. Feature values can be perturbed by replacing original values with values from an equidistant grid of that feature, with permuted or randomly subsampled values [19], or with quantiles. We highlight two major issues: First, if features are dependent, all three perturbation approaches produce unrealistic data points, i.e. the new data points are located outside of the multivariate joint distribution of the data (see Fig. 4). Second, even if features are independent, using an equidistant grid can produce unrealistic values for the feature of interest. Consider a feature that follows a skewed distribution with outliers. An equidistant grid would generate many values between outliers and non-outliers. In contrast to the grid-based approach, the other two approaches maintain the marginal distribution of the feature of interest.

Both issues can result in misleading interpretations (illustrative examples are given in [55,84]), since the model is evaluated in areas of the feature space with few or no observed real data points, where model uncertainty can be expected to be very high. This issue is aggravated if interpretation methods integrate over such points with the same weight and confidence as for much more realistic samples with high model confidence.

Solution: Before applying interpretation methods, practitioners should check for dependencies between features in the data, e.g. via descriptive statistics or measures of dependence (see Sect. 5.2). When it is unavoidable to include dependent features in the model (which is usually the case in ML scenarios), additional information regarding the strength and shape of the dependence structure should be provided. Sometimes, alternative interpretation methods can be used as a workaround or to provide additional information. Accumulated local effect plots (ALE) [3] can be applied when features are dependent, but can produce non-intuitive effect plots for simple linear models with interactions [45]. For other methods such as the PFI, conditional variants exist [17,84,107]. In the case of LIME, it was suggested to focus in sampling on realistic (i.e. close to the data manifold) [97] and relevant areas (e.g. close to the decision boundary) [69]. Note, however, that conditional interpretations are often different and should not be used as a substitute for unconditional interpretations (see Sect. 5.3). Furthermore, dependent features should not be interpreted separately but rather jointly. This can be achieved by visualizing e.g. a 2-dimensional ALE plot of two dependent features, which, admittedly, only works for very low-dimensional combinations. Especially in high-dimensional settings where dependent features

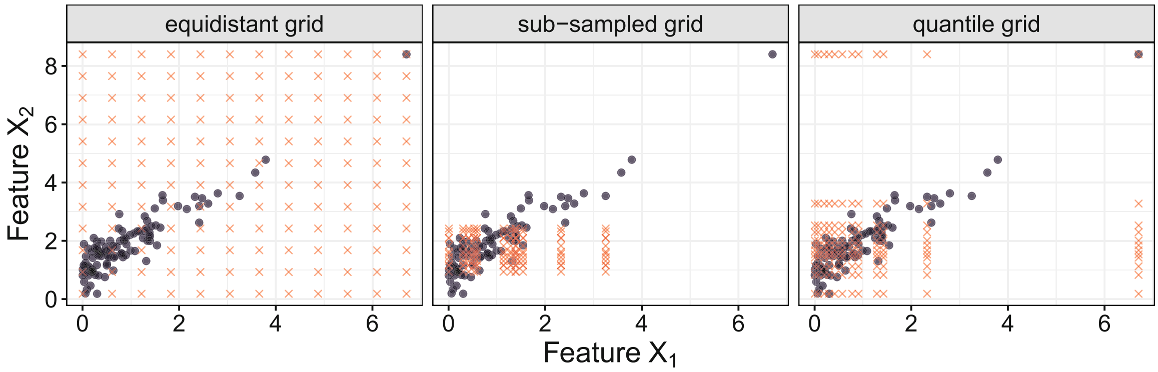


Fig. 4. Interpretation with extrapolation. Illustration of artificial data points generated by three different perturbation approaches. The black dots refer to observed data points and the red crosses to the artificial data points.

can be grouped in a meaningful way, grouped interpretation methods might be more reasonable (see Sect. 9.1).

We recommend using quantiles or randomly subsampled values over equidistant grids. By default, many implementations of interpretability methods use an equidistant grid to perturb feature values [41,81,89], although some also allow using user-defined values.

Open Issues: A comprehensive comparison of strategies addressing extrapolation and how they affect an interpretation method is currently missing. This also includes studying interpretation methods and their conditional variants when they are applied to data with different dependence structures.

5.2 Confusing Linear Correlation with General Dependence

Pitfall: Features with a Pearson correlation coefficient (PCC) close to zero can still be dependent and cause misleading model interpretations (see Fig. 5). While independence between two features implies that the PCC is zero, the converse is generally false. The PCC, which is often used to analyze dependence, only tracks linear correlations and has other shortcomings such as sensitivity to outliers [113]. Any type of dependence between features can have a strong impact on the interpretation of the results of IML methods (see Sect. 5.1). Thus, knowledge about the (possibly non-linear) dependencies between features is crucial for an informed use of IML methods.

Solution: Low-dimensional data can be visualized to detect dependence (e.g. scatter plots) [80]. For high-dimensional data, several other measures of dependence in addition to PCC can be used. If dependence is monotonic, Spearman's rank correlation coefficient [72] can be a simple, robust alternative to PCC. For categorical or mixed features, separate dependence measures have been proposed, such as Kendall's rank correlation coefficient for ordinal features, or the phi coefficient and Goodman & Kruskal's lambda for nominal features [59].

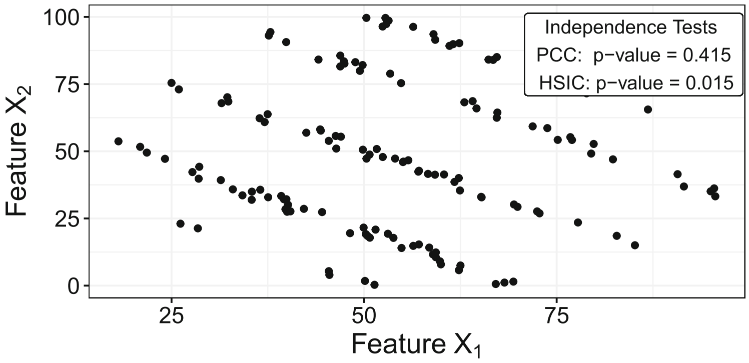


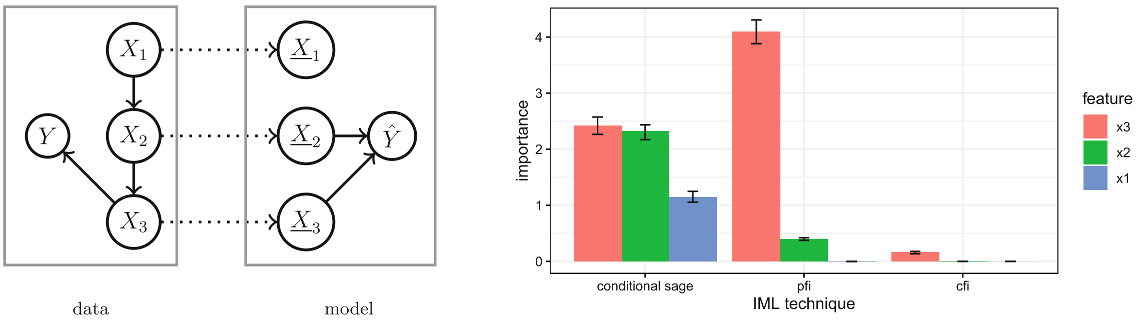
Fig. 5. Confusing linear correlation with dependence. Highly dependent features X_1 and X_2 that have a correlation close to zero. A test (H_0: Features are independent) using Pearson correlation is not significant, but for HSIC, the H_0-hypothesis gets rejected. Data from [80].

Studying non-linear dependencies is more difficult since a vast variety of possible associations have to be checked. Nevertheless, several non-linear association measures with sound statistical properties exist. Kernel-based measures, such as kernel canonical correlation analysis (KCCA) [6] or the Hilbert-Schmidt independence criterion (HSIC) [44], are commonly used. They have a solid theoretical foundation, are computationally feasible, and robust [113]. In addition, there are information-theoretical measures, such as (conditional) mutual information [24] or the maximal information coefficient (MIC) [93], that can however be difficult to estimate [9,116]. Other important measures are e.g. the distance correlation [111], the randomized dependence coefficient (RDC) [74], or the alternating conditional expectations (ACE) algorithm [14]. In addition to using PCC, we recommend using at least one measure that detects non-linear dependencies (e.g. HSIC).

5.3 Misunderstanding Conditional Interpretation

Pitfall: Conditional variants of interpretation techniques avoid extrapolation but require a different interpretation. Interpretation methods that perturb features independently of others will extrapolate under dependent features but provide insight into the model's mechanism [56,61]. Therefore, these methods are said to be true to the model but not true to the data [21].

For feature effect methods such as the PDP, the plot can be interpreted as the isolated, average effect the feature has on the prediction. For the PFI, the importance can be interpreted as the drop in performance when the feature's information is "destroyed" (by perturbing it). Marginal SHAP value functions [78] quantify a feature's contribution to a specific prediction, and marginal SAGE value functions [25] quantify a feature's contribution to the overall prediction performance. All the aforementioned methods extrapolate under dependent features (see also Sect. 5.1), but satisfy sensitivity, i.e. are zero if a feature is not used by the model [25,56,61,110].



(a) (b)

Fig. 6. Misunderstanding conditional interpretation. A linear model was fitted on the data-generating process modeled using a linear Gaussian structural causal model. The entailed directed acyclic graph is depicted on the left. For illustrative purposes, the original model coefficients were updated such that not only feature X_3, but also feature X_2 is used by the model. PFI on test data considers both X_3 and X_2 to be relevant. In contrast, conditional feature importance variants either only consider X_3 to be relevant (CFI) or consider all features to be relevant (conditional SAGE value function).

Conditional variants of these interpretation methods do not replace feature values independently of other features, but in such a way that they conform to the conditional distribution. This changes the interpretation as the effects of all dependent features become entangled. Depending on the method, conditional sampling leads to a more or less restrictive notion of relevance.

For example, for dependent features, the Conditional Feature Importance (CFI) [17,84,107,117] answers the question: "How much does the model performance drop if we permute a feature, *but given that we know the values of the other features?*" [63,84,107].[1] Two highly dependent features might be individually important (based on the unconditional PFI), but have a very low conditional importance score because the information of one feature is contained in the other and vice versa.

In contrast, the conditional variant of PDP, called marginal plot or M-plot [3], violates sensitivity, i.e. may even show an effect for features that are not used by the model. This is because for M-plots, the feature of interest is not sampled conditionally on the remaining features, but rather the remaining features are sampled conditionally on the feature of interest. As a consequence, the distribution of dependent covariates varies with the value of the feature of interest. Similarly, conditional SAGE and conditional SHAP value functions sample the remaining features conditional on the feature of interest and therefore violate sensitivity [25,56,61,109].

We demonstrate the difference between PFI, CFI, and conditional SAGE value functions on a simulated example (Fig. 6) where the data-generating mech-

[1] While for CFI the conditional independence of the feature of interest X_j with the target Y given the remaining features X_{-j} ($Y \perp X_j | X_{-j}$) is already a sufficient condition for zero importance, the corresponding PFI may still be nonzero [63].

anism is known. While PFI only considers features to be relevant if they are actually used by the model, SAGE value functions may also consider a feature to be important that is not directly used by the model if it contains information that the model exploits. CFI only considers a feature to be relevant if it is both mechanistically used by the model and contributes unique information about Y.

Solution: When features are highly dependent and conditional effects and importance scores are used, the practitioner must be aware of the distinct interpretation. Recent work formalizes the implications of marginal and conditional interpretation techniques [21,25,56,61,63]. While marginal methods provide insight into the model's mechanism but are not true to the data, their conditional variants are not true to the model but provide insight into the associations in the data.

If joint insight into model and data is required, designated methods must be used. ALE plots [3] provide interval-wise unconditional interpretations that are true to the data. They have been criticized to produce non-intuitive results for certain data-generating mechanisms [45]. Molnar et al. [84] propose a subgroup-based conditional sampling technique that allows for group-wise marginal interpretations that are true to model and data and that can be applied to feature importance and feature effects methods such as conditional PDPs and CFI. For feature importance, the DEDACT framework [61] allows to decompose conditional importance measures such as SAGE value functions into their marginal contributions and vice versa, thereby allowing global insight into both: the sources of prediction-relevant information in the data as well as into the feature pathways by which the information enters the model.

Open Issues: The quality of conditional IML techniques depends on the goodness of the conditional sampler. Especially in continuous, high-dimensional settings, conditional sampling is challenging. More research on the robustness of interpretation techniques regarding the quality of the sample is required.

6 Misleading Interpretations Due to Feature Interactions

6.1 Misleading Feature Effects Due to Aggregation

Pitfall: Global interpretation methods, such as PDP or ALE plots, visualize the average effect of a feature on a model's prediction. However, they can produce misleading interpretations when features interact. Figure 7 A and B show the marginal effect of features X_1 and X_2 of the below-stated simulation example. While the PDP of the non-interacting feature X_1 seems to capture the true underlying effect of X_1 on the target quite well (A), the global aggregated effect of the interacting feature X_2 (B) shows almost no influence on the target, although an effect is clearly there by construction.

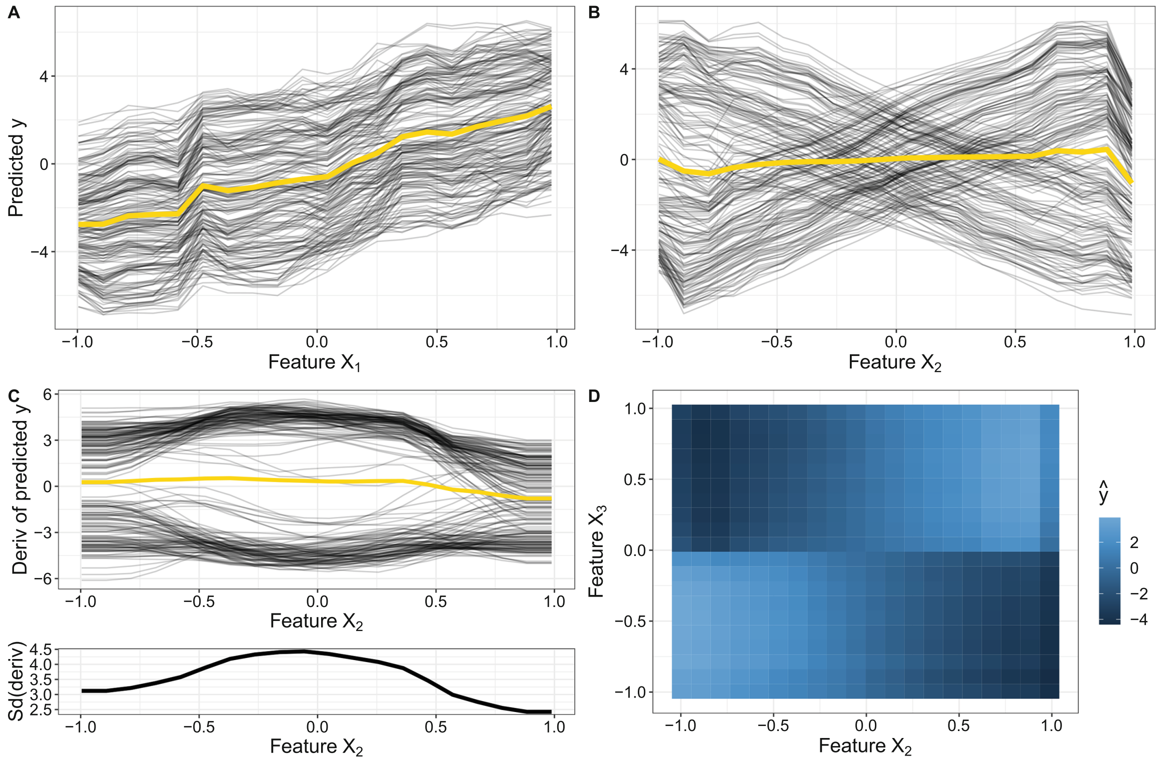


Fig. 7. Misleading effect due to interactions. Simulation example with interactions: $Y = 3X_1 - 6X_2 + 12X_2\mathbb{1}_{(X_3 \geq 0)} + \epsilon$ with $X_1, X_2, X_3 \overset{i.i.d.}{\sim} U[-1, 1]$ and $\epsilon \overset{i.i.d.}{\sim} N(0, 0.3)$. A random forest with 500 trees is fitted on 1000 observations. Effects are calculated on 200 randomly sampled (training) observations. **A, B:** PDP (yellow) and ICE curves of X_1 and X_2; **C:** Derivative ICE curves and their standard deviation of X_2; **D:** 2-dimensional PDP of X_2 and X_3.

Solution: For the PDP, we recommend to additionally consider the corresponding ICE curves [38]. While PDP and ALE average out interaction effects, ICE curves directly show the heterogeneity between individual predictions. Figure 7 A illustrates that the individual marginal effect curves all follow an upward trend with only small variations. Hence, by aggregating these ICE curves to a global marginal effect curve such as the PDP, we do not lose much information. However, when the regarded feature interacts with other features, such as feature X_2 with feature X_3 in this example, then marginal effect curves of different observations might not show similar effects on the target. Hence, ICE curves become very heterogeneous, as shown in Fig. 7 B. In this case, the influence of feature X_2 is not well represented by the global average marginal effect. Particularly for continuous interactions where ICE curves start at different intercepts, we recommend the use of derivative or centered ICE curves, which eliminate differences in intercepts and leave only differences due to interactions [38]. Derivative ICE curves also point out the regions of highest interaction with other features. For example, Fig. 7 C indicates that predictions for X_2 taking values close to 0 strongly depend on other features' values. While these methods show that interactions are present with regards to the feature of interest but do not reveal other features with which it interacts, the 2-dimensional PDP or ALE plot are options to visualize 2-way interaction effects. The 2-dimensional PDP in Fig. 7 D shows that predictions with regards to feature X_2 highly depend on the feature values of feature X_3.

Other methods that aim to gain more insights into these visualizations are based on clustering homogeneous ICE curves, such as visual interaction effects (VINE) [16] or [122]. As an example, in Fig. 7 B, it would be more meaningful to average over the upward and downward proceeding ICE curves separately and hence show that the average influence of feature X_2 on the target depends on an interacting feature (here: X_3). Work by Zon et al. [125] followed a similar idea by proposing an interactive visualization tool to group Shapley values with regards to interacting features that need to be defined by the user.

Open Issues: The introduced visualization methods are not able to illustrate the type of the underlying interaction and most of them are also not applicable to higher-order interactions.

6.2 Failing to Separate Main from Interaction Effects

Pitfall: Many interpretation methods that quantify a feature's importance or effect cannot separate an interaction from main effects. The PFI, for example, includes both the importance of a feature and the importance of all its interactions with other features [19]. Also local explanation methods such as LIME and Shapley values only provide additive explanations without separation of main effects and interactions [40].

Solution: Functional ANOVA introduced by [53] is probably the most popular approach to decompose the joint distribution into main and interaction effects. Using the same idea, the H-Statistic [35] quantifies the interaction strength between two features or between one feature and all others by decomposing the 2-dimensional PDP into its univariate components. The H-Statistic is based on the fact that, in the case of non-interacting features, the 2-dimensional partial dependence function equals the sum of the two underlying univariate partial dependence functions. Another similar interaction score based on partial dependencies is defined by [42]. Instead of decomposing the partial dependence function, [87] uses the predictive performance to measure interaction strength. Based on Shapley values, Lundberg et al. [77] proposed SHAP interaction values, and Casalicchio et al. [19] proposed a fair attribution of the importance of interactions to the individual features.

Furthermore, Hooker [54] considers dependent features and decomposes the predictions in main and interaction effects. A way to identify higher-order interactions is shown in [53].

Open Issues: Most methods that quantify interactions are not able to identify higher-order interactions and interactions of dependent features. Furthermore, the presented solutions usually lack automatic detection and ranking of all interactions of a model. Identifying a suitable shape or form of the modeled interaction is not straightforward as interactions can be very different and complex, e.g., they can be a simple product of features (multiplicative interaction) or can have a complex joint non-linear effect such as smooth spline surface.

7 Ignoring Model and Approximation Uncertainty

Pitfall: Many interpretation methods only provide a mean estimate but do not quantify uncertainty. Both the model training and the computation of interpretation are subject to uncertainty. The model is trained on (random) data, and therefore should be regarded as a random variable. Similarly, LIME's surrogate model relies on perturbed and reweighted samples of the data to approximate the prediction function locally [94]. Other interpretation methods are often defined in terms of expectations over the data (PFI, PDP, Shapley values, ...), but are approximated using Monte Carlo integration. Ignoring uncertainty can result in the interpretation of noise and non-robust results. The true effect of a feature may be flat, but – purely by chance, especially on smaller datasets – the Shapley value might show an effect. This effect could cancel out once averaged over multiple model fits.

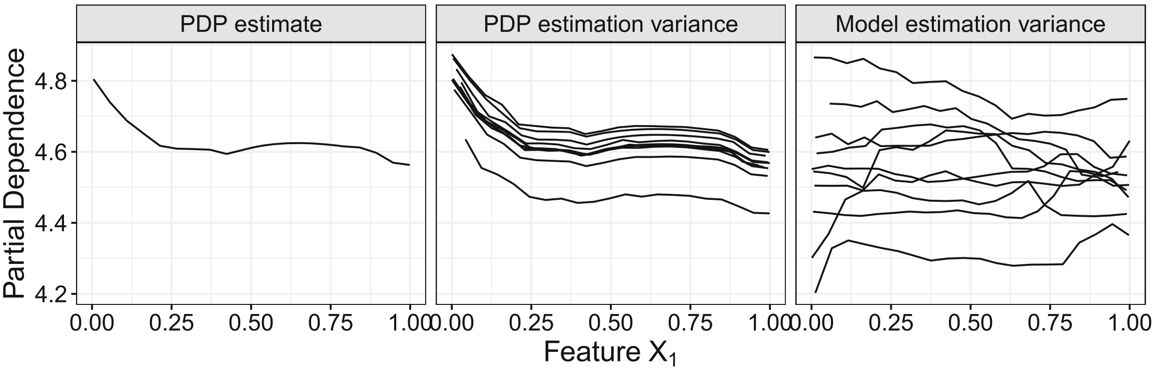


Fig. 8. Ignoring model and approximation uncertainty. PDP for X_1 with $Y = 0 \cdot X_1 + \sum_{j=2}^{10} X_j + \epsilon_i$ with $X_1, \ldots, X_{10} \sim U[0,1]$ and $\epsilon_i \sim N(0, 0.9)$. **Left:** PDP for X_1 of a random forest trained on 100 data points. **Middle:** Multiple PDPs (10x) for the model from left plots, but with different samples (each n=100) for PDP estimation. **Right:** Repeated (10x) data samples of n=100 and newly fitted random forest.

Figure 8 shows that a single PDP (first plot) can be misleading because it does not show the variance due to PDP estimation (second plot) and model fitting (third plot). If we are not interested in learning about a specific model, but rather about the relationship between feature X_1 and the target (in this case), we should consider the model variance.

Solution: By repeatedly computing PDP and PFI with a given model, but with different permutations or bootstrap samples, the uncertainty of the estimate can be quantified, for example in the form of confidence intervals. For PFI, frameworks for confidence intervals and hypothesis tests exist [2,117], but they assume a fixed model. If the practitioner wants to condition the analysis on the modeling process and capture the process' variance instead of conditioning on a fixed model, PDP and PFI should be computed on multiple model fits [83].

Open Issues: While Moosbauer et al. [85] derived confidence bands for PDPs for probabilistic ML models that cover the model's uncertainty, a general model-agnostic uncertainty measure for feature effect methods such as ALE [3] and PDP [36] has (to the best of our knowledge) not been introduced yet.

8 Ignoring the Rashomon Effect

Pitfall: Sometimes different models explain the data-generating process equally well, but contradict each other. This phenomenon is called the Rashomon effect, named after the movie "Rashomon" from the year 1950. Breiman formalized it for predictive models in 2001 [13]: Different prediction models might perform equally well (Rashomon set), but construct the prediction function in a different way (e.g. relying on different features). This can result in conflicting interpretations and conclusions about the data. Even small differences in the training data can cause one model to be preferred over another.

For example, Dong and Rudin [29] identified a Rashomon set of equally well performing models for the COMPAS dataset. They showed that the models differed greatly in the importance they put on certain features. Specifically, if criminal history was identified as less important, race was more important and vice versa. Cherry-picking one model and its underlying explanation might not be sufficient to draw conclusions about the data-generating process. As Hancox-Li [48] states "just because race happens to be an unimportant variable in that one explanation does not mean that it is objectively an unimportant variable".

The Rashomon effect can also occur at the level of the interpretation method itself. Differing hyperparameters or interpretation goals can be one reason (see Sect. 2). But even if the hyperparameters are fixed, we could still obtain contradicting explanations by an interpretation method, e.g., due to a different data sample or initial seed.

A concrete example of the Rashomon effect is counterfactual explanations. Different counterfactuals may all alter the prediction in the desired way, but point to different feature changes required for that change. If a person is deemed uncreditworthy, one corresponding counterfactual explaining this decision may point to a scenario in which the person had asked for a shorter loan duration and amount, while another counterfactual may point to a scenario in which the person had a higher income and more stable job. Focusing on only one counterfactual explanation in such cases strongly limits the possible epistemic access.

Solution: If multiple, equally good models exist, their interpretations should be compared. Variable importance clouds [29] is a method for exploring variable importance scores for equally good models within one model class. If the interpretations are in conflict, conclusions must be drawn carefully. Domain experts or further constraints (e.g. fairness or sparsity) could help to pick a suitable model. Semenova et al. [102] also hypothesized that a large Rashomon set could contain simpler or more interpretable models, which should be preferred according to Sect. 4.

In the case of counterfactual explanations, multiple, equally good explanations exist. Here, methods that return a set of explanations rather than a single one should be used – for example, the method by Dandl et al. [26] or Mothilal et al. [86].

Open Issues: Numerous very different counterfactual explanations are overwhelming for users. Methods for aggregating or combining explanations are still a matter of future research.

9 Failure to Scale to High-Dimensional Settings

9.1 Human-Intelligibility of High-Dimensional IML Output

Pitfall: Applying IML methods naively to high-dimensional datasets (e.g. visualizing feature effects or computing importance scores on feature level) leads to an overwhelming and high-dimensional IML output, which impedes human analysis. Especially interpretation methods that are based on visualizations make it difficult for practitioners in high-dimensional settings to focus on the most important insights.

Solution: A natural approach is to reduce the dimensionality before applying any IML methods. Whether this facilitates understanding or not depends on the possible semantic interpretability of the resulting, reduced feature space – as features can either be selected or dimensionality can be reduced by linear or non-linear transformations. Assuming that users would like to interpret in the original feature space, many feature selection techniques can be used [46], resulting in much sparser and consequently easier to interpret models. Wrapper selection approaches are model-agnostic and algorithms like greedy forward selection or subset selection procedures [5,60], which start from an empty model and iteratively add relevant (subsets of) features if needed, even allow to measure the relevance of features for predictive performance. An alternative is to directly use models that implicitly perform feature selection such as LASSO [112] or component-wise boosting [99] as they can produce sparse models with fewer features. In the case of LIME or other interpretation methods based on surrogate models, the aforementioned techniques could be applied to the surrogate model.

When features can be meaningfully grouped in a data-driven or knowledge-driven way [51], applying IML methods directly to grouped features instead of single features is usually more time-efficient to compute and often leads to more appropriate interpretations. Examples where features can naturally be grouped include the grouping of sensor data [20], time-lagged features [75], or one-hot-encoded categorical features and interaction terms [43]. Before a model is fitted, groupings could already be exploited for dimensionality reduction, for example by selecting groups of features by the group LASSO [121].

For model interpretation, various papers extended feature importance methods from single features to groups of features [5,43,114,119]. In the case of grouped PFI, this means that we perturb the entire group of features at once and measure the performance drop compared to the unperturbed dataset. Compared to standard PFI, the grouped PFI does not break the association to the other features of the group, but to features of other groups and the target. This is especially useful when features within the same group are highly correlated (e.g. time-lagged features), but between-group dependencies are rather low. Hence, this might also be a possible solution for the extrapolation pitfall described in Sect. 5.1.

We consider the PhoneStudy in [106] as an illustration. The PhoneStudy dataset contains 1821 features to analyze the link between human behavior based on smartphone data and participants' personalities. Interpreting the results in this use case seems to be challenging since features were dependent and single feature effects were either small or non-linear [106]. The features have been grouped in behavior-specific categories such as app-usage, music consumption, or overall phone usage. Au et al. [5] calculated various grouped importance scores on the feature groups to measure their influence on a specific personality trait (e.g. conscientiousness). Furthermore, the authors applied a greedy forward subset selection procedure via repeated subsampling on the feature groups and showed that combining app-usage features and overall phone usage features were most of the times sufficient for the given prediction task.

Open Issues: The quality of a grouping-based interpretation strongly depends on the human intelligibility and meaningfulness of the grouping. If the grouping structure is not naturally given, then data-driven methods can be used. However, if feature groups are not meaningful (e.g. if they cannot be described by a super-feature such as app-usage), then subsequent interpretations of these groups are purposeless. One solution could be to combine feature selection strategies with interpretation methods. For example, LIME's surrogate model could be a LASSO model. However, beyond surrogate models, the integration of feature selection strategies remains an open issue that requires further research.

Existing research on grouped interpretation methods mainly focused on quantifying grouped feature importance, but the question of "how a group of features influences a model's prediction" remains almost unanswered. Only recently, [5,15,101] attempted to answer this question by using dimension-reduction techniques (such as PCA) before applying the interpretation method. However, this is also a matter of further research.

9.2 Computational Effort

Pitfall: Some interpretation methods do not scale linearly with the number of features. For example, for the computation of exact Shapley values the number of possible coalitions [25,78], or for a (full) functional ANOVA decomposition the number of components (main effects plus all interactions) scales with $\mathcal{O}(2^p)$ [54].[2]

Solution: For the functional ANOVA, a common solution is to keep the analysis to the main effects and selected 2-way interactions (similar for PDP and ALE). Interesting 2-way interactions can be selected by another method such as the H-statistic [35]. However, the selection of 2-way interactions requires additional computational effort. Interaction strength usually decreases quickly with increasing interaction size, and one should only consider d-way interactions when all their $(d-1)$-way interactions were significant [53]. For Shapley-based methods, an efficient approximation exists that is based on randomly sampling and evaluating feature orderings until the estimates converge. The variance of the estimates reduces in $\mathcal{O}(\frac{1}{m})$, where m is the number of evaluated orderings [25,78].

9.3 Ignoring Multiple Comparison Problem

Pitfall: Simultaneously testing the importance of multiple features will result in false-positive interpretations if the multiple comparisons problem (MCP) is ignored. The MCP is well known in significance tests for linear models and exists similarly in testing for feature importance in ML. For example, suppose we simultaneously test the importance of 50 features (with the H_0-hypothesis of zero importance) at the significance level $\alpha = 0.05$. Even if all features are unimportant, the probability of observing that at least one feature is significantly important is $1 - \mathbb{P}(\text{‘no feature important’}) = 1 - (1 - 0.05)^{50} \approx 0.923$. Multiple comparisons become even more problematic the higher the dimension of the dataset.

Solution: Methods such as Model-X knockoffs [17] directly control for the false discovery rate (FDR). For all other methods that provide p-values or confidence intervals, such as PIMP (Permutation IMPortance) [2], which is a testing approach for PFI, MCP is often ignored in practice to the best of our knowledge, with some exceptions[105,117]. One of the most popular MCP adjustment methods is the Bonferroni correction [31], which rejects a null hypothesis if its p-value is smaller than α/p, with p as the number of tests. It has the disadvantage that it increases the probability of false negatives [90]. Since MCP is well known in statistics, we refer the practitioner to [28] for an overview and discussion of alternative adjustment methods, such as the Bonferroni-Holm method [52].

[2] Similar to the PDP or ALE plots, the functional ANOVA components describe individual feature effects and interactions.

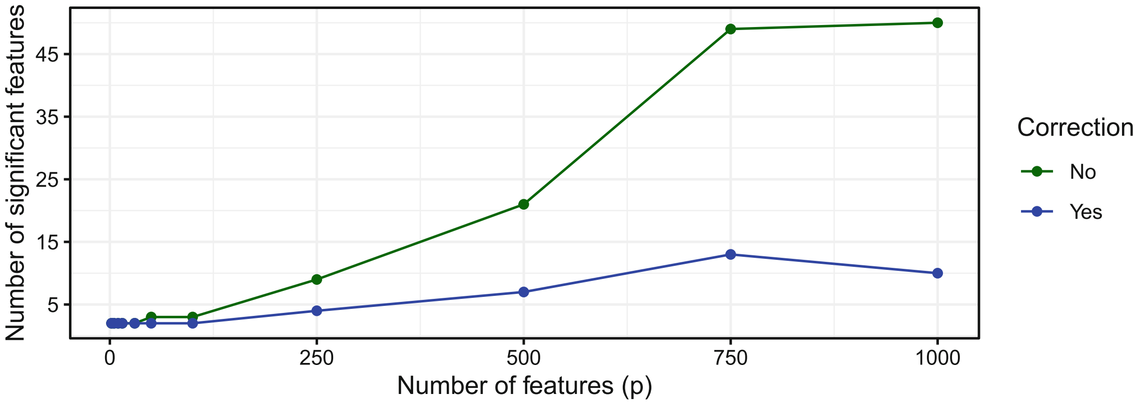


Fig. 9. Failure to scale to high-dimensional settings. Comparison of the number of features with significant importance - once with and once without Bonferroni-corrected significance levels for a varying number of added noise variables. Datasets were sampled from $Y = 2X_1 + 2X_2^2 + \epsilon$ with $X_1, X_2, \epsilon \sim N(0,1)$. $X_3, X_4, ..., X_p \sim N(0,1)$ are additional noise variables with p ranging between 2 and 1000. For each p, we sampled two datasets from this data-generating process – one to train a random forest with 500 trees on and one to test whether feature importances differed from 0 using PIMP. In all experiments, X_1 and X_2 were correctly identified as important.

As an example, in Fig. 9 we compare the number of features with significant importance measured by PIMP once with and once without Bonferroni-adjusted significance levels ($\alpha = 0.05$ vs. $\alpha = 0.05/p$). Without correcting for multi-comparisons, the number of features mistakenly evaluated as important grows considerably with increasing dimension, whereas Bonferroni correction results in only a modest increase.

10 Unjustified Causal Interpretation

Pitfall: Practitioners are often interested in causal insights into the underlying data-generating mechanisms, which IML methods do not generally provide. Common causal questions include the identification of causes and effects, predicting the effects of interventions, and answering counterfactual questions [88]. For example, a medical researcher might want to identify risk factors or predict average and individual treatment effects [66]. In search of answers, a researcher can therefore be tempted to interpret the result of IML methods from a causal perspective.

However, a causal interpretation of predictive models is often not possible. Standard supervised ML models are not designed to model causal relationships but to merely exploit associations. A model may therefore rely on causes and effects of the target variable as well as on variables that help to reconstruct unobserved influences on Y, e.g. causes of effects [118]. Consequently, the question of whether a variable is relevant to a predictive model (indicated e.g. by PFI > 0) does not directly indicate whether a variable is a cause, an effect, or does not stand in any causal relation to the target variable. Furthermore, even if a model would rely solely on direct causes for the prediction, the causal structure between features must be taken into account. Intervening on a variable in the real world may affect not only Y but also other variables in the feature set. Without assumptions about the underlying causal structure, IML methods cannot account for these adaptions and guide action [58,62].

As an example, we constructed a dataset by sampling from a structural causal model (SCM), for which the corresponding causal graph is depicted in Fig. 10. All relationships are linear Gaussian with variance 1 and coefficients 1. For a linear model fitted on the dataset, all features were considered to be relevant based on the model coefficients ($\hat{y} = 0.329x_1 + 0.323x_2 - 0.327x_3 + 0.342x_4 + 0.334x_5$, $R^2 = 0.943$), although x_3, x_4 and x_5 do not cause Y.

Solution: The practitioner must carefully assess whether sufficient assumptions can be made about the underlying data-generating process, the learned model, and the interpretation technique. If these assumptions are met, a causal interpretation may be possible. The PDP between a feature and the target can be interpreted as the respective average causal effect if the model performs well and the set of remaining variables is a valid adjustment set [123]. When it is known whether a model is deployed in a causal or anti-causal setting – i.e. whether the model attempts to predict an effect from its causes or the other way round – a partial identification of the causal roles based on feature relevance is possible (under strong and non-testable assumptions) [118]. Designated tools and approaches are available for causal discovery and inference [91].

Open Issues: The challenge of causal discovery and inference remains an open key issue in the field of ML. Careful research is required to make explicit under which assumptions what insight about the underlying data-generating mechanism can be gained by interpreting an ML model.

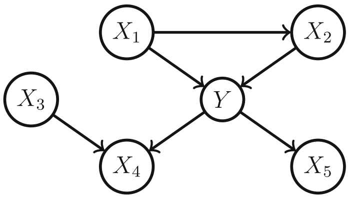


Fig. 10. Causal graph

11 Discussion

In this paper, we have reviewed numerous pitfalls of local and global model-agnostic interpretation techniques, e.g. in the case of bad model generalization, dependent features, interactions between features, or causal interpretations. We have not attempted to provide an exhaustive list of all potential pitfalls in ML model interpretation, but have instead focused on common pitfalls that apply to various model-agnostic IML methods and pose a particularly high risk.

We have omitted pitfalls that are more specific to one IML method type: For local methods, the vague notions of neighborhood and distance can lead to misinterpretations [68,69], and common distance metrics (such as the Euclidean distance) are prone to the curse of dimensionality [1]; Surrogate methods such as LIME may not be entirely faithful to the original model they replace in interpretation. Moreover, we have not addressed pitfalls associated with certain data types (like the definition of superpixels in image data [98]), nor those related to human cognitive biases (e.g. the illusion of model understanding [22]).

Many pitfalls in the paper are strongly linked with axioms that encode desiderata of model interpretation. For example, pitfall Sect. 5.3 (misunderstanding conditional interpretations) is related to violations of sensitivity [56,110]. As such, axioms can help to make the strengths and limitations of methods explicit. Therefore, we encourage an axiomatic evaluation of interpretation methods.

We hope to promote a more cautious approach when interpreting ML models in practice, to point practitioners to already (partially) available solutions, and to stimulate further research on these issues. The stakes are high: ML algorithms are increasingly used for socially relevant decisions, and model interpretations play an important role in every empirical science. Therefore, we believe that users can benefit from concrete guidance on properties, dangers, and problems of IML techniques – especially as the field is advancing at high speed. We need to strive towards a recommended, well-understood set of tools, which will in turn require much more careful research. This especially concerns the meta-issues of comparisons of IML techniques, IML diagnostic tools to warn against misleading interpretations, and tools for analyzing multiple dependent or interacting features.

References

1. Aggarwal, C.C., Hinneburg, A., Keim, D.A.: On the surprising behavior of distance metrics in high dimensional space. In: Van den Bussche, J., Vianu, V. (eds.) ICDT 2001. LNCS, vol. 1973, pp. 420–434. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44503-X_27
2. Altmann, A., Toloşi, L., Sander, O., Lengauer, T.: Permutation importance: a corrected feature importance measure. Bioinformatics **26**(10), 1340–1347 (2010). https://doi.org/10.1093/bioinformatics/btq134
3. Apley, D.W., Zhu, J.: Visualizing the effects of predictor variables in black box supervised learning models. J. R. Stat. Soc.: Ser. B (Stat. Methodol.) **82**(4), 1059–1086 (2020). https://doi.org/10.1111/rssb.12377
4. Arlot, S., Celisse, A.: A survey of cross-validation procedures for model selection. Statist. Surv. **4**, 40–79 (2010). https://doi.org/10.1214/09-SS054
5. Au, Q., Herbinger, J., Stachl, C., Bischl, B., Casalicchio, G.: Grouped feature importance and combined features effect plot. arXiv preprint arXiv:2104.11688 (2021)
6. Bach, F.R., Jordan, M.I.: Kernel independent component analysis. J. Mach. Learn. Res. **3**(Jul), 1–48 (2002)

7. Baesens, B., Van Gestel, T., Viaene, S., Stepanova, M., Suykens, J., Vanthienen, J.: Benchmarking state-of-the-art classification algorithms for credit scoring. J. Oper. Res. Soc. **54**(6), 627–635 (2003). https://doi.org/10.1057/palgrave.jors.2601545
8. Bansal, N., Agarwal, C., Nguyen, A.: SAM: the sensitivity of attribution methods to hyperparameters. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8673–8683 (2020)
9. Belghazi, M.I., et al.: Mutual information neural estimation. In: International Conference on Machine Learning, pp. 531–540 (2018)
10. Bischl, B., et al.: Hyperparameter optimization: foundations, algorithms, best practices and open challenges. arXiv preprint arXiv:2107.05847 (2021)
11. Bischl, B., Mersmann, O., Trautmann, H., Weihs, C.: Resampling methods for meta-model validation with recommendations for evolutionary computation. Evol. Comput. **20**(2), 249–275 (2012). https://doi.org/10.1162/EVCO_a_00069
12. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001). https://doi.org/10.1023/A:1010933404324
13. Breiman, L.: Statistical modeling: the two cultures (with comments and a rejoinder by the author). Stat. Sci. **16**(3), 199–231 (2001). https://doi.org/10.1214/ss/1009213726
14. Breiman, L., Friedman, J.H.: Estimating optimal transformations for multiple regression and correlation. J. Am. Stat. Assoc. **80**(391), 580–598 (1985). https://doi.org/10.1080/01621459.1985.10478157
15. Brenning, A.: Transforming feature space to interpret machine learning models. arXiv:2104.04295 (2021)
16. Britton, M.: Vine: visualizing statistical interactions in black box models. arXiv preprint arXiv:1904.00561 (2019)
17. Candes, E., Fan, Y., Janson, L., Lv, J.: Panning for gold:'model-x'knockoffs for high dimensional controlled variable selection. J. R. Stat. Soc.: Ser. B (Stat. Methodol.) **80**(3), 551–577 (2018). https://doi.org/10.1111/rssb.12265
18. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., Elhadad, N.: Intelligible models for healthcare: predicting pneumonia risk and hospital 30-day readmission. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1721–1730 (2015). https://doi.org/10.1145/2783258.2788613
19. Casalicchio, G., Molnar, C., Bischl, B.: Visualizing the feature importance for black box models. In: Berlingerio, M., Bonchi, F., Gärtner, T., Hurley, N., Ifrim, G. (eds.) ECML PKDD 2018. LNCS (LNAI), vol. 11051, pp. 655–670. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-10925-7_40
20. Chakraborty, D., Pal, N.R.: Selecting useful groups of features in a connectionist framework. IEEE Trans. Neural Netw. **19**(3), 381–396 (2008). https://doi.org/10.1109/TNN.2007.910730
21. Chen, H., Janizek, J.D., Lundberg, S., Lee, S.I.: True to the model or true to the data? arXiv preprint arXiv:2006.16234 (2020)
22. Chromik, M., Eiband, M., Buchner, F., Krüger, A., Butz, A.: I think I get your point, AI! the illusion of explanatory depth in explainable AI. In: 26th International Conference on Intelligent User Interfaces, IUI 2021, pp. 307–317. Association for Computing Machinery, New York (2021). https://doi.org/10.1145/3397481.3450644
23. Claeskens, G., Hjort, N.L., et al.: Model Selection and Model Averaging. Cambridge Books (2008). https://doi.org/10.1017/CBO9780511790485

24. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley (2012). https://doi.org/10.1002/047174882X
25. Covert, I., Lundberg, S.M., Lee, S.I.: Understanding global feature contributions with additive importance measures. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 17212–17223. Curran Associates, Inc. (2020)
26. Dandl, S., Molnar, C., Binder, M., Bischl, B.: Multi-objective counterfactual explanations. In: Bäck, T., et al. (eds.) PPSN 2020. LNCS, vol. 12269, pp. 448–469. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58112-1_31
27. Das, A., Rad, P.: Opportunities and challenges in explainable artificial intelligence (XAI): a survey. arXiv preprint arXiv:2006.11371 (2020)
28. Dickhaus, T.: Simultaneous Statistical Inference. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-45182-9
29. Dong, J., Rudin, C.: Exploring the cloud of variable importance for the set of all good models. Nat. Mach. Intell. **2**(12), 810–824 (2020). https://doi.org/10.1038/s42256-020-00264-0
30. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608 (2017)
31. Dunn, O.J.: Multiple comparisons among means. J. Am. Stat. Assoc. **56**(293), 52–64 (1961). https://doi.org/10.1080/01621459.1961.10482090
32. Fernández-Delgado, M., Cernadas, E., Barro, S., Amorim, D.: Do we need hundreds of classifiers to solve real world classification problems. J. Mach. Learn. Res. **15**(1), 3133–3181 (2014). https://doi.org/10.5555/2627435.2697065
33. Fisher, A., Rudin, C., Dominici, F.: All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously. J. Mach. Learn. Res. **20**(177), 1–81 (2019)
34. Freiesleben, T.: Counterfactual explanations & adversarial examples-common grounds, essential differences, and potential transfers. arXiv preprint arXiv:2009.05487 (2020)
35. Friedman, J.H., Popescu, B.E.: Predictive learning via rule ensembles. Ann. Appl. Stat. **2**(3), 916–954 (2008). https://doi.org/10.1214/07-AOAS148
36. Friedman, J.H., et al.: Multivariate adaptive regression splines. Ann. Stat. **19**(1), 1–67 (1991). https://doi.org/10.1214/aos/1176347963
37. Garreau, D., von Luxburg, U.: Looking deeper into tabular lime. arXiv preprint arXiv:2008.11092 (2020)
38. Goldstein, A., Kapelner, A., Bleich, J., Pitkin, E.: Peeking inside the black box: visualizing statistical learning with plots of individual conditional expectation. J. Comput. Graph. Stat. **24**(1), 44–65 (2015). https://doi.org/10.1080/10618600.2014.907095
39. Good, P.I., Hardin, J.W.: Common Errors in Statistics (and How to Avoid Them). Wiley (2012). https://doi.org/10.1002/9781118360125
40. Gosiewska, A., Biecek, P.: Do not trust additive explanations. arXiv preprint arXiv:1903.11420 (2019)
41. Greenwell, B.M.: PDP: an R package for constructing partial dependence plots. R J. **9**(1), 421–436 (2017). https://doi.org/10.32614/RJ-2017-016
42. Greenwell, B.M., Boehmke, B.C., McCarthy, A.J.: A simple and effective model-based variable importance measure. arXiv:1805.04755 (2018)
43. Gregorutti, B., Michel, B., Saint-Pierre, P.: Grouped variable importance with random forests and application to multiple functional data analysis. Comput. Stat. Data Anal. **90**, 15–35 (2015). https://doi.org/10.1016/j.csda.2015.04.002

44. Gretton, A., Bousquet, O., Smola, A., Schölkopf, B.: Measuring statistical dependence with Hilbert-Schmidt norms. In: Jain, S., Simon, H.U., Tomita, E. (eds.) ALT 2005. LNCS (LNAI), vol. 3734, pp. 63–77. Springer, Heidelberg (2005). https://doi.org/10.1007/11564089_7
45. Grömping, U.: Model-agnostic effects plots for interpreting machine learning models. Reports in Mathematics, Physics and Chemistry Report 1/2020 (2020)
46. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**(Mar), 1157–1182 (2003)
47. Hall, P.: On the art and science of machine learning explanations. arXiv preprint arXiv:1810.02909 (2018)
48. Hancox-Li, L.: Robustness in machine learning explanations: does it matter? In: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, FAT* 2020, pp. 640–647. Association for Computing Machinery, New York (2020). https://doi.org/10.1145/3351095.3372836
49. Hand, D.J.: Classifier technology and the illusion of progress. Stat. Sci. **21**(1), 1–14 (2006). https://doi.org/10.1214/088342306000000060
50. Hastie, T., Tibshirani, R.: Generalized additive models. Stat. Sci. **1**(3), 297–310 (1986). https://doi.org/10.1214/ss/1177013604
51. He, Z., Yu, W.: Stable feature selection for biomarker discovery. Comput. Biol. Chem. **34**(4), 215–225 (2010). https://doi.org/10.1016/j.compbiolchem.2010.07.002
52. Holm, S.: A simple sequentially rejective multiple test procedure. Scand. J. Stat. **6**(2), 65–70 (1979)
53. Hooker, G.: Discovering additive structure in black box functions. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2004, pp. 575–580. Association for Computing Machinery, New York (2004). https://doi.org/10.1145/1014052.1014122
54. Hooker, G.: Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables. J. Comput. Graph. Stat. **16**(3), 709–732 (2007). https://doi.org/10.1198/106186007X237892
55. Hooker, G., Mentch, L.: Please stop permuting features: an explanation and alternatives. arXiv preprint arXiv:1905.03151 (2019)
56. Janzing, D., Minorics, L., Blöbaum, P.: Feature relevance quantification in explainable AI: a causality problem. arXiv preprint arXiv:1910.13413 (2019)
57. Kadir, T., Brady, M.: Saliency, scale and image description. Int. J. Comput. Vis. **45**(2), 83–105 (2001). https://doi.org/10.1023/A:1012460413855
58. Karimi, A.H., Schölkopf, B., Valera, I.: Algorithmic recourse: from counterfactual explanations to interventions. arXiv:2002.06278 (2020)
59. Khamis, H.: Measures of association: how to choose? J. Diagn. Med. Sonography **24**(3), 155–162 (2008). https://doi.org/10.1177/8756479308317006
60. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artif. Intell. **97**(1–2), 273–324 (1997)
61. König, G., Freiesleben, T., Bischl, B., Casalicchio, G., Grosse-Wentrup, M.: Decomposition of global feature importance into direct and associative components (DEDACT). arXiv preprint arXiv:2106.08086 (2021)
62. König, G., Freiesleben, T., Grosse-Wentrup, M.: A causal perspective on meaningful and robust algorithmic recourse. arXiv preprint arXiv:2107.07853 (2021)
63. König, G., Molnar, C., Bischl, B., Grosse-Wentrup, M.: Relative feature importance. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 9318–9325. IEEE (2021). https://doi.org/10.1109/ICPR48806.2021.9413090

64. Krishnan, M.: Against interpretability: a critical examination of the interpretability problem in machine learning. Philos. Technol. **33**(3), 487–502 (2019). https://doi.org/10.1007/s13347-019-00372-9
65. Kuhle, S., et al.: Comparison of logistic regression with machine learning methods for the prediction of fetal growth abnormalities: a retrospective cohort study. BMC Pregnancy Childbirth **18**(1), 1–9 (2018). https://doi.org/10.1186/s12884-018-1971-2
66. König, G., Grosse-Wentrup, M.: A Causal Perspective on Challenges for AI in Precision Medicine (2019)
67. Lang, M., et al.: MLR3: a modern object-oriented machine learning framework in R. J. Open Source Softw. (2019). https://doi.org/10.21105/joss.01903
68. Laugel, T., Lesot, M.J., Marsala, C., Renard, X., Detyniecki, M.: The dangers of post-hoc interpretability: unjustified counterfactual explanations. In: Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, pp. 2801–2807. International Joint Conferences on Artificial Intelligence Organization (2019)
69. Laugel, T., Renard, X., Lesot, M.J., Marsala, C., Detyniecki, M.: Defining locality for surrogates in post-hoc interpretablity. arXiv preprint arXiv:1806.07498 (2018)
70. Lauritsen, S.M., et al.: Explainable artificial intelligence model to predict acute critical illness from electronic health records. Nat. Commun. **11**(1), 1–11 (2020). https://doi.org/10.1038/s41467-020-17431-x
71. Lessmann, S., Baesens, B., Seow, H.V., Thomas, L.C.: Benchmarking state-of-the-art classification algorithms for credit scoring: an update of research. Eur. J. Oper. Res. **247**(1), 124–136 (2015). https://doi.org/10.1016/j.ejor.2015.05.030
72. Liebetrau, A.: Measures of Association. No. Bd. 32; Bd. 1983 in 07, SAGE Publications (1983)
73. Lipton, Z.C.: The mythos of model interpretability. Queue **16**(3), 31–57 (2018). https://doi.org/10.1145/3236386.3241340
74. Lopez-Paz, D., Hennig, P., Schölkopf, B.: The randomized dependence coefficient. In: Advances in Neural Information Processing Systems, pp. 1–9 (2013). https://doi.org/10.5555/2999611.2999612
75. Lozano, A.C., Abe, N., Liu, Y., Rosset, S.: Grouped graphical granger modeling for gene expression regulatory networks discovery. Bioinformatics **25**(12), i110–i118 (2009). https://doi.org/10.1093/bioinformatics/btp199
76. Lundberg, S.M., et al.: From local explanations to global understanding with explainable AI for trees. Nat. Mach. Intell. **2**(1), 56–67 (2020). https://doi.org/10.1038/s42256-019-0138-9
77. Lundberg, S.M., Erion, G.G., Lee, S.I.: Consistent individualized feature attribution for tree ensembles. arXiv preprint arXiv:1802.03888 (2018)
78. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: NIPS, vol. 30, pp. 4765–4774. Curran Associates, Inc. (2017). https://doi.org/10.5555/3295222.3295230
79. Makridakis, S., Spiliotis, E., Assimakopoulos, V.: Statistical and machine learning forecasting methods: concerns and ways forward. PloS One **13**(3) (2018). https://doi.org/10.1371/journal.pone.0194889
80. Matejka, J., Fitzmaurice, G.: Same stats, different graphs: generating datasets with varied appearance and identical statistics through simulated annealing. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp. 1290–1294 (2017). https://doi.org/10.1145/3025453.3025912

81. Molnar, C., Casalicchio, G., Bischl, B.: IML: an R package for interpretable machine learning. J. Open Source Softw. **3**(26), 786 (2018). https://doi.org/10.21105/joss.00786
82. Molnar, C., Casalicchio, G., Bischl, B.: Quantifying model complexity via functional decomposition for better post-hoc interpretability. In: Cellier, P., Driessens, K. (eds.) ECML PKDD 2019. CCIS, vol. 1167, pp. 193–204. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-43823-4_17
83. Molnar, C., Freiesleben, T., König, G., Casalicchio, G., Wright, M.N., Bischl, B.: Relating the partial dependence plot and permutation feature importance to the data generating process. arXiv preprint arXiv:2109.01433 (2021)
84. Molnar, C., König, G., Bischl, B., Casalicchio, G.: Model-agnostic feature importance and effects with dependent features-a conditional subgroup approach. arXiv preprint arXiv:2006.04628 (2020)
85. Moosbauer, J., Herbinger, J., Casalicchio, G., Lindauer, M., Bischl, B.: Towards explaining hyperparameter optimization via partial dependence plots. In: 8th ICML Workshop on Automated Machine Learning (AutoML) (2020)
86. Mothilal, R.K., Sharma, A., Tan, C.: Explaining machine learning classifiers through diverse counterfactual explanations. CoRR abs/1905.07697 (2019). http://arxiv.org/abs/1905.07697
87. Oh, S.: Feature interaction in terms of prediction performance. Appl. Sci. **9**(23) (2019). https://doi.org/10.3390/app9235191
88. Pearl, J., Mackenzie, D.: The Ladder of Causation. The Book of Why: The New Science of Cause and Effect, pp. 23–52. Basic Books, New York (2018). https://doi.org/10.1080/14697688.2019.1655928
89. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011). https://doi.org/10.5555/1953048.2078195
90. Perneger, T.V.: What's wrong with Bonferroni adjustments. BMJ **316**(7139), 1236–1238 (1998). https://doi.org/10.1136/bmj.316.7139.1236
91. Peters, J., Janzing, D., Scholkopf, B.: Elements of Causal Inference - Foundations and Learning Algorithms. The MIT Press (2017). https://doi.org/10.5555/3202377
92. Philipp, M., Rusch, T., Hornik, K., Strobl, C.: Measuring the stability of results from supervised statistical learning. J. Comput. Graph. Stat. **27**(4), 685–700 (2018). https://doi.org/10.1080/10618600.2018.1473779
93. Reshef, D.N., et al.: Detecting novel associations in large data sets. Science **334**(6062), 1518–1524 (2011). https://doi.org/10.1126/science.1205438
94. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should I trust you?: explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144. ACM (2016). https://doi.org/10.1145/2939672.2939778
95. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nat. Mach. Intell. **1**(5), 206–215 (2019). https://doi.org/10.1038/s42256-019-0048-x
96. Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., Zhong, C.: Interpretable machine learning: fundamental principles and 10 grand challenges. arXiv preprint arXiv:2103.11251 (2021)
97. Saito, S., Chua, E., Capel, N., Hu, R.: Improving lime robustness with smarter locality sampling. arXiv preprint arXiv:2006.12302 (2020)
98. Schallner, L., Rabold, J., Scholz, O., Schmid, U.: Effect of superpixel aggregation on explanations in lime-a case study with biological data. arXiv preprint arXiv:1910.07856 (2019)

99. Schmid, M., Hothorn, T.: Boosting additive models using component-wise p-splines. Comput. Stat. Data Anal. **53**(2), 298–311 (2008). https://doi.org/10.1016/j.csda.2008.09.009
100. Scholbeck, C.A., Molnar, C., Heumann, C., Bischl, B., Casalicchio, G.: Sampling, intervention, prediction, aggregation: a generalized framework for model-agnostic interpretations. In: Cellier, P., Driessens, K. (eds.) ECML PKDD 2019. CCIS, vol. 1167, pp. 205–216. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-43823-4_18
101. Seedorff, N., Brown, G.: Totalvis: a principal components approach to visualizing total effects in black box models. SN Comput. Sci. **2**(3), 1–12 (2021). https://doi.org/10.1007/s42979-021-00560-5
102. Semenova, L., Rudin, C., Parr, R.: A study in Rashomon curves and volumes: a new perspective on generalization and model simplicity in machine learning. arXiv preprint arXiv:1908.01755 (2021)
103. Shalev-Shwartz, S., Ben-David, S.: Understanding Machine Learning: From Theory to Algorithms. Cambridge University Press, Cambridge (2014)
104. Simon, R.: Resampling strategies for model assessment and selection. In: Dubitzky, W., Granzow, M., Berrar, D. (eds.) Fundamentals of Data Mining in Genomics and Proteomics, pp. 173–186. Springer, Cham (2007). https://doi.org/10.1007/978-0-387-47509-7_8
105. Stachl, C., et al.: Behavioral patterns in smartphone usage predict big five personality traits. PsyArXiv (2019). https://doi.org/10.31234/osf.io/ks4vd
106. Stachl, C., et al.: Predicting personality from patterns of behavior collected with smartphones. Proc. Natl. Acad. Sci. (2020). https://doi.org/10.1073/pnas.1920484117
107. Strobl, C., Boulesteix, A.L., Kneib, T., Augustin, T., Zeileis, A.: Conditional variable importance for random forests. BMC Bioinform. **9**(1), 307 (2008). https://doi.org/10.1186/1471-2105-9-307
108. Štrumbelj, E., Kononenko, I.: Explaining prediction models and individual predictions with feature contributions. Knowl. Inf. Syst. **41**(3), 647–665 (2013). https://doi.org/10.1007/s10115-013-0679-x
109. Sundararajan, M., Najmi, A.: The many Shapley values for model explanation. arXiv preprint arXiv:1908.08474 (2019)
110. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: International Conference on Machine Learning, pp. 3319–3328. PMLR (2017)
111. Székely, G.J., Rizzo, M.L., Bakirov, N.K., et al.: Measuring and testing dependence by correlation of distances. Ann. Stat. **35**(6), 2769–2794 (2007). https://doi.org/10.1214/009053607000000505
112. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. Roy. Stat. Soc.: Ser. B (Methodol.) **58**(1), 267–288 (1996). https://doi.org/10.1111/j.1467-9868.2011.00771.x
113. Tjøstheim, D., Otneim, H., Støve, B.: Statistical dependence: beyond pearson's p. arXiv preprint arXiv:1809.10455 (2018)
114. Valentin, S., Harkotte, M., Popov, T.: Interpreting neural decoding models using grouped model reliance. PLoS Comput. Biol. **16**(1), e1007148 (2020). https://doi.org/10.1371/journal.pcbi.1007148
115. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: automated decisions and the GDPR. Harv. JL Tech. **31**, 841 (2017). https://doi.org/10.2139/ssrn.3063289

116. Walters-Williams, J., Li, Y.: Estimation of mutual information: a survey. In: Wen, P., Li, Y., Polkowski, L., Yao, Y., Tsumoto, S., Wang, G. (eds.) RSKT 2009. LNCS (LNAI), vol. 5589, pp. 389–396. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02962-2_49
117. Watson, D.S., Wright, M.N.: Testing conditional independence in supervised learning algorithms. arXiv preprint arXiv:1901.09917 (2019)
118. Weichwald, S., Meyer, T., Özdenizci, O., Schölkopf, B., Ball, T., Grosse-Wentrup, M.: Causal interpretation rules for encoding and decoding models in neuroimaging. Neuroimage **110**, 48–59 (2015). https://doi.org/10.1016/j.neuroimage.2015.01.036
119. Williamson, B.D., Gilbert, P.B., Simon, N.R., Carone, M.: A unified approach for inference on algorithm-agnostic variable importance. arXiv:2004.03683 (2020)
120. Wu, J., Roy, J., Stewart, W.F.: Prediction modeling using EHR data: challenges, strategies, and a comparison of machine learning approaches. Med. Care S106–S113 (2010). https://doi.org/10.1097/MLR.0b013e3181de9e17
121. Yuan, M., Lin, Y.: Model selection and estimation in regression with grouped variables. J. R. Stat. Soc.: Ser. B (Statistical Methodology) **68**(1), 49–67 (2006). https://doi.org/10.1111/j.1467-9868.2005.00532.x
122. Zhang, X., Wang, Y., Li, Z.: Interpreting the black box of supervised learning models: visualizing the impacts of features on prediction. Appl. Intell. **51**(10), 7151–7165 (2021). https://doi.org/10.1007/s10489-021-02255-z
123. Zhao, Q., Hastie, T.: Causal interpretations of black-box models. J. Bus. Econ. Stat. 1–10 (2019). https://doi.org/10.1080/07350015.2019.1624293
124. Zhao, X., Lovreglio, R., Nilsson, D.: Modelling and interpreting pre-evacuation decision-making using machine learning. Autom. Constr. **113**, 103140 (2020). https://doi.org/10.1016/j.autcon.2020.103140
125. van der Zon, S.B., Duivesteijn, W., van Ipenburg, W., Veldsink, J., Pechenizkiy, M.: ICIE 1.0: a novel tool for interactive contextual interaction explanations. In: Alzate, C., et al. (eds.) MIDAS/PAP -2018. LNCS (LNAI), vol. 11054, pp. 81–94. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-13463-1_6

2.6 Paper VI: Efficient SAGE Estimation via Causal Structure Learning

Luther*, Christoph, Gunnar König*, Moritz Grosse-Wentrup. **Efficient SAGE Estimation via Causal Structure Learning.** *International Conference on Artificial Intelligence and Statistics.* PMLR, 2023.

Gunnar König contributed to the paper as shared first author. Gunnar König had the initial idea, proved Theorem 1, wrote the feature importance code, did the real-world application and supervised Christoph Luther during his Master's thesis. Gunnar König and Christoph designed the experiments together. Christoph Luther conducted the experiments, visualized the results and wrote large parts of the paper. Gunnar König revised the structure of the article, contributing paragraphs to all sections. All authors helped to edit and proofread the paper.

Efficient SAGE Estimation via Causal Structure Learning

Christoph Luther*
University of Vienna
UniVie Doctoral School CS

Gunnar König*
LMU Munich
University of Vienna
Munich Center for ML (MCML)

Moritz Grosse-Wentrup
University of Vienna
Data Science @ Uni Vienna
Vienna CogSciHub

Abstract

The Shapley Additive Global Importance (SAGE) value is a theoretically appealing interpretability method that fairly attributes global importance to a model's features. However, its exact calculation requires the computation of the feature's surplus performance contributions over an exponential number of feature sets. This is computationally expensive, particularly because estimating the surplus contributions requires sampling from conditional distributions. Thus, SAGE approximation algorithms only take a fraction of the feature sets into account. We propose d-SAGE, a method that accelerates SAGE approximation. d-SAGE is motivated by the observation that conditional independencies (CIs) between a feature and the model target imply zero surplus contributions, such that their computation can be skipped. To identify CIs, we leverage causal structure learning (CSL) to infer a graph that encodes (conditional) independencies in the data as d-separations. This is computationally more efficient because the expense of the one-time graph inference and the d-separation queries is negligible compared to the expense of surplus contribution evaluations. Empirically we demonstrate that d-SAGE enables the efficient and accurate estimation of SAGE values.

1 INTRODUCTION

Machine learning (ML) is increasingly deployed in various fields, ranging from the sciences (Reichstein et al., 2019; Schmidt et al., 2019; Luan and Tsai, 2021; Farrell et al., 2018) to high-stakes decisions about individuals (Raghavan et al., 2020; Zeng et al., 2017; Obermeyer and Mullainathan, 2019). Despite impressive successes in predictive performance (Senior et al., 2020; Bhatt et al., 2020), the complexity of ML models makes it difficult to assess their trustworthiness or to gain knowledge about the data generating process. In recent years, the advent of interpretable machine learning has brought about a plethora of methods that provide insight into model and data (Molnar, 2020). Among those, interpretability methods based on the Shapley value from game theory (Shapley, 1953) have gained popularity as they satisfy desirable fairness properties (Štrumbelj and Kononenko, 2014; Datta et al., 2016; Lundberg and Lee, 2017; Sundararajan and Najmi, 2020; Covert et al., 2020).

SAGE values (Covert et al., 2020) apply Shapley values to fairly attribute the model's predictive performance to the features, thereby providing valuable insight into dependencies in the data. They are particularly appealing for scientific inference since they can be linked to properties of the data generating process (Covert et al., 2020; Freiesleben et al., 2022). The building blocks for SAGE values are so-called SAGE value functions $\nu(\mathbf{X}_S)$ that measure the performance contribution of arbitrary subsets of features $\mathbf{X}_S$. Based on these value functions, a feature's importance value ϕ is computed as the average surplus contribution $\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S)$ of the feature X_j over all possible subsets $\mathbf{X}_S$ of the remaining features. This is a computationally demanding procedure due to the number of coalitions $\mathbf{X}_S$ that grows exponentially with the number of features (Covert et al., 2020; Van den Broeck et al., 2022) and the high expense of evaluating ν which stems from the conditional sampling that is required for its estimation. In practice, (Covert et al., 2020) address the exponential number of coalitions by only computing the respective surplus contribution for a randomly sampled subset of the coalitions.[1]

*Equal contribution.

Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS) 2023, Valencia, Spain. PMLR: Volume 206. Copyright 2023 by the author(s).

[1] Furthermore, Covert et al. (2020) avoid conditional sampling for the evaluation of ν by employing marginal sampling instead. If features are dependent, this leads to extrapolation and does not allow linking the SAGE values to properties of the data generating process (Chen et al., 2020). In this work, we focus on estimating conditional SAGE values.

In this work, we suggest exploiting the dependence structure in the data to speed up the estimation of (conditional sampling based) SAGE values in an approach we coin d-SAGE. More specifically, we show that the surplus contribution $\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S)$ is zero for optimal predictors if the variable of interest is conditionally independent of the model's target given the respective subset of remaining features (i.e., if $X_j \perp Y|\mathbf{X}_S$, Theorem 1). As such, if we know the conditional independencies (CIs) in the data, the respective value function evaluations can be skipped. Since, in general, the dependence structure is unknown, and conditional independence testing is expensive, we leverage research in causal structure learning (CSL) that allows us to greedily learn graphical models which encode the dependence structure in the data.

Overall, the approach is based on the following rationale: The quality of SAGE approximation hinges on the number of evaluations of ν that each require estimating conditional expectations and thus are computationally expensive.[2] d-SAGE relies on the one-time estimation of a causal graph, which in practice can be performed by greedy-search algorithms in polynomial time (Scutari et al., 2019b). The estimated graph then allows to identify CIs using linear-time d-separation queries (Hagberg et al., 2008; Darwiche, 2009). Every found d-separation, in turn, warrants to spare an expensive evaluation of $\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S)$. Since graph learning has to be performed only once and d-separation queries are highly efficient, the runtime of SAGE estimation can be reduced significantly by skipping the computation of $\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S)$ whenever warranted. We show empirically that the saved runtime is approximately equal to the share of CIs.

1.1 Contributions

We propose d-SAGE, the first method that exploits the dependence structure in the data to make SAGE estimation more efficient. More specifically, we find that CIs in the data imply that the respective (expensive) surplus evaluations can be skipped and suggest leveraging greedy CSL for their identification (Section 4). To select a suitable CSL algorithm, we perform a benchmark that, in contrast to previous work, evaluates the algorithms' ability to efficiently identify CIs in the data (Section 5.1). On twelve synthetic datasets, we demonstrate empirically that d-SAGE and the approximation algorithm by Covert et al. (2020) converge towards the same estimates but that d-SAGE is significantly faster. We find that the computational overhead of learning the causal structure is negligible compared to the computational cost of the surplus evaluations, such that the overall runtime reduction is approximately equal to the share of CIs found in the data (Section 5.2). Consequently, d-SAGE enables the application of SAGE for larger models, especially in sparse settings.

2 RELATED WORK

While there are many attempts to tackle the complexity of Shapley value based methods, most existing work targets speeding up SHAP (Lundberg and Lee, 2017) estimation (Jethani et al., 2021; Covert and Lee, 2021; Li et al., 2020) or is limited to be applied with random forests (Bénard et al., 2022). In contrast, our work is model-agnostic and targets improving SAGE estimation. Moreover, none of the existing work exploits the dependence structure in the data to yield efficiency gains. As such, we see our work as complementary to the approach of Mitchell et al. (2022), who suggest to carefully select permutations.

In recent years, concepts from causality have also been introduced to Shapley value based importance measures to adapt them to answer specific questions or to improve model interpretation. Frye et al. (2020b), for example, introduce *asymmetric Shapley values* that can either shift the explanatory power of all variables along a causal chain towards the root cause (distal approach) or towards immediate causes (proximate approach). Moreover, Heskes et al. (2020) use Pearl's do-calculus to develop *causal Shapley values* and Wang et al. (2021) propose to attach importance to edges in a causal graph instead of explanatory variables, i.e., nodes in the graph. In contrast to the literature, we seek efficiency gains for feature attributions from causal inference research while retaining the principle of SAGE values unaltered.

We do, however, make use of CSL. Scutari et al. (2019a) and Constantinou et al. (2021) provide large-scale benchmark studies of structure learning algorithms. In short, both studies agree on the superiority of score-based structure learning based on greedy search algorithms over constraint-based and hybrid methods. These findings motivate our choice of CSL algorithms for d-separation inference. In contrast to existing work, our benchmark does not focus on recovering the causal structure but on detecting CIs in the data.

3 BACKGROUND

This section serves to familiarise the reader with the basic concepts required to understand this paper. First, we introduce SAGE values for global feature importance. We then explain CSL, which we later use to speed up SAGE estimation.

[2] The expense of the computation depends on the type of data for which the conditional expectation shall be computed. Previous work in the field assumes polynomial complexity for the operation (Van den Broeck et al., 2022).

3.1 Shapley Additive Global Importance

The Shapley value, which was initially proposed in game theory (Shapley, 1953), is commonly applied for feature relevance quantification (Štrumbelj and Kononenko, 2014; Datta et al., 2016; Lundberg and Lee, 2017; Sundararajan and Najmi, 2020; Covert et al., 2020). In the study of cooperative games, it serves to fairly attribute the outcome of a game to all participating players. The principle can be applied to assess the relevance of variables for a predictor f, where the predictive performance is the outcome of the game and the variables are the players. Covert et al. (2020) leverage Shapley values to derive a *global* measure of feature importance, i.e. SAGE values. Global in this context means that the importance of a feature across all instances in a sample is assessed. For an arbitrary model $\hat{f}$ using inputs $x_1, ..., x_d$, Covert et al. (2020) define the SAGE value for the j-th feature as:

$$\begin{aligned}\phi_j(\nu) = \frac{1}{d!} \sum_{\pi \in \Pi(d)} \big(&\nu(\{X_i : \pi(i) \leq \pi(j)\}) \\ &- \nu(\{X_i : \pi(i) < \pi(j)\})\big)\end{aligned} \tag{1}$$

where X_j is the random variable corresponding to feature observation x_j, $\Pi(d)$ is the set of all permutations of indices $\{1, ..., d\}$, π a specific permutation and $\pi(j)$ the position of feature j in permutation π. For the sake of readability, we use the more general notation $\mathbf{X}_S$ instead of $\{X_i : \pi(i) < \pi(j)\}$ as input to the value function ν with $\mathbf{X}_S$ being any set of features and S the collection of indices of the contained features, i.e. $S \subseteq \{1, ..., d\}$ ($\bar{S}$ is its complementary set). $\nu(\mathbf{X}_S)$ is defined as

$$\nu(\mathbf{X}_S) = \mathbb{E}_{\mathbf{X},Y}[\ell(\hat{f}_\emptyset(\mathbf{X}_\emptyset), Y)] - \mathbb{E}_{\mathbf{X},Y}[\ell(\hat{f}_S(\mathbf{X}_S), Y)],$$

where $\ell(\cdot)$ is any admissible loss function and $\hat{f}_S(\mathbf{x}_S) = \mathbb{E}_{\mathbf{X}_{\bar{S}}|\mathbf{X}_S}[\hat{f}(\mathbf{X})|\mathbf{X}_S = \mathbf{x}_S]$. Thus, $\nu(\mathbf{X}_S)$ is the reduction in risk induced by adding $\mathbf{X}_S$. Consequently, SAGE values gauge a feature j's importance using the average over the additional reduction in risk of the feature compared to any existing coalition.

SAGE values are particularly appealing as they satisfy six desirable fairness axioms that set them apart from other feature importance measures: *efficiency*, the *dummy property*, *symmetry*, *monotonicity*, *linearity*[3] and invariance to monotone transformations. Despite a thorough mathematical foundation and the fulfilment of mentioned desiderata, SAGE values have a major drawback: They require the evaluation of an exponential number of surplus evaluations, which is computationally infeasible. In practice, only a subset of possible coalitions is evaluated (cf. Section 3.2).

To estimate SAGE values, access to the conditional feature distributions is required; More specifically, we need to sample from $P(\mathbf{X}_{\bar{S}}|\mathbf{X}_S)$ to estimate the marginalized prediction $\hat{f}_S(\mathbf{x}_S) = \mathbb{E}_{\mathbf{X}_{\bar{S}}|\mathbf{X}_S}[\hat{f}(\mathbf{X})|\mathbf{X}_S = \mathbf{x}_S]$. However, conditional samplers may not be readily available in practice. Covert et al. (2020) suggest eluding the problem by sampling from $P(\mathbf{X}_{\bar{S}})$ instead (marginal sampling). Albeit easy to implement (and computationally efficient), marginal sampling may generate unrealistic data points $(\mathbf{x}_S, \mathbf{x}_{\bar{S}})$ and thus marginal-sampling based SAGE values are not suitable for inference about the data generating process or to understand the model's behaviour in the observational distribution (Frye et al., 2020a; Chen et al., 2020; Aas et al., 2021; Molnar et al., 2022). Therefore, we focus on conditional SAGE and estimate the conditional distributions if they are not known.

For conditional distribution estimation, a variety of techniques can be employed. For categorical variables, estimating the conditional reduces to standard supervised learning with cross-entropy loss. For linear Gaussian data, it can be estimated analytically from the covariance matrix (Page Jr, 1984). A range of methods exist for continuous settings with nonlinearities (Bishop, 1994; Bashtannyk and Hyndman, 2001; Sohn et al., 2015; Trippe and Turner, 2018; Winkler et al., 2019; Hothorn and Zeileis, 2021). For mixed data, a sequential design can be employed (Blesch et al., 2022).

3.2 Intractability of SAGE and Approximation Algorithm

For the Shapley based interpretability approach SHAP intractability was proven (Van den Broeck et al., 2022). For the exact computation, the surplus contribution for all possible subsets of the remaining features must be evaluated. The number of possible subsets grows exponentially in the number of features.

Exact SAGE estimation also suffers from the exponential number of coalitions. To address the issue, Covert et al. (2020) propose an approximation algorithm that does not take all possible coalitions into account. More specifically, the authors propose to repetitively sample permutations π from the feature indices. Then, for every element of the current permutation, starting with the first one, they successively compute $\Delta_{j|S} := \nu(\mathbf{X}_{S \cup j}) - \nu(\mathbf{X}_S)$ with the set $\mathbf{X}_S$ being all features that come before the feature of interest j in π. The mean of all $\Delta_{j|S}$ values for X_j over the number of repetitions then is its estimated importance $\hat{\phi}_j(\nu)$. The approximation algorithm is unbiased and the variance of the estimate reduces in $O(\frac{1}{n})$ (Covert et al., 2020). However, considering the risk evaluation required for estimating ν, the procedure based on conditional sampling remains computationally demanding.

[3] For simplicity we employ the names of these Shapley value properties for the SAGE properties that are described in Appendix D.

3.3 Causal Structure Learning

This section deals with the introduction of CSL used to estimate graphs representing d-separations. d-separation is the graphical equivalent to conditional independence in the underlying distribution. Both concepts are indeed equivalent under two standard assumptions: (1) that the Markov property is fulfilled and (2) that the distribution is faithful w.r.t. the graph. Since we merely use graphs to read off d-separations, we leave out a holistic coverage and refer the reader to Darwiche (2009) and Pearl (2009). Here, it shall suffice that we refer to a directed acyclic graph (DAG) whose nodes represent random variables from the underlying distribution and whose edges reflect direct dependencies in the data. Edge directions are further interpreted as cause-effect relations. We now briefly summarise the inference of such graphs from data.

Generally, one distinguishes between *constraint-based* and *score-based* methods. The former use CIs inferred from data as constraints on where to draw edges. The latter explore the space of all possible DAGs over the given variables and assign scores to every visited graph. The output of the algorithm is the highest scoring graph. Since the space of DAGs over a set of variables or nodes grows super-exponentially in the set's cardinality, score-based methods often rely on greedy search techniques. In addition, *hybrid methods* combine both CIs as constraints and scoring of graphs to assess candidates.

In this work, we focus on greedy structure learning that performed best in recent benchmarks (Scutari et al., 2019a; Constantinou et al., 2021). More precisely, we rely on structure inference based on hill-climbing (HC) and TABU search (Russell and Norvig, 2009; Scutari et al., 2019b). Crucially, both algorithms use the Bayesian information criterion (Schwarz, 1978), which satisfies two key properties, *consistency* and *local consistency*[4] (Gámez et al., 2011; Chickering, 2003). Gámez et al. (2011) show that for HC for a dataset of size n and *iid* data, the output graph is a minimal I-Map of the underlying distribution if $n \to \infty$ and the scoring function satisfies *consistency* and *local consistency*. By definition of a minimal I-Map, the set of CIs represented by d-separation in the graph is a subset of the CIs in the distribution. Hence, while there might be independencies in the underlying distribution of the data not represented by d-separation, there are no instances of d-separations that do not correspond to independencies. Note that HC introduces a DAG structure of the output graph but the assumption on the data is just being an *iid* sample. The proof, however, hinges on the assumption of faithfulness. For linear models, though, the probability of faithfulness being violated is shown to be zero if model parameters are randomly drawn from positive densities (cf. Peters et al. (2017), Spirtes et al. (2000)). While there is no similar theoretical result for TABU, the latter is an extension of HC and exhibits similar behaviour in practice (cf. Section 5.1).

4 CAUSAL STRUCTURE LEARNING FOR EFFICIENT SAGE ESTIMATION

SAGE estimation is computationally challenging. For an exact computation, the surplus contribution of the feature of interest j with respect to every possible coalition $\mathbf{X}_S$ of the remaining features must be computed. The surplus contribution is defined as in Section 3.2

$$\Delta_{j|S} = \nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S) \tag{2}$$

The number of possible coalitions grows exponentially in the number of features, making the exact computation intractable in high-dimensional settings. SAGE values are therefore estimated by randomly sampling coalitions until the estimates converge (Section 3.2). Nevertheless, estimation remains challenging since evaluating $\Delta_{j|S}$ requires sampling from conditional distributions, and therefore even one evaluation is a significant computational challenge. Thus, in practice, the approximation quality is limited by the number of surplus contributions that can be computed.

We propose d-SAGE, an approach that can identify and skip unnecessary surplus evaluations and thereby allows to improve the approximation quality. The method is based on the observation that $\Delta_{j|S}$ evaluates to zero if X_j is conditionally independent of Y given X_S:

Theorem 1. *For ℓ being cross-entropy loss or the mean-squared error, f^* the respective optimal predictor and ν_{ℓ,f^*} the corresponding SAGE value function, it holds that*

$$X_j \perp Y|\boldsymbol{X}_S \Rightarrow \nu_{\ell,f^*}(\boldsymbol{X}_{S\cup j}) - \nu_{\ell,f^*}(\boldsymbol{X}_S) = 0.$$

Proof (sketch, full proof in A): Covert et al. (2020) show that for the cross entropy loss function with its respective optimal model, the Bayes classifier, Equation 2 equals the conditional mutual information of X_j and Y given $\boldsymbol{X}_S$, i.e. $I(X_j;Y|\boldsymbol{X}_S)$. A similar result holds for optimal regression models with the mean squared error (MSE) as loss function. In this case, the surplus contribution is shown to be equal to $\mathbb{E}_{\boldsymbol{X}_S}[Var(\mathbb{E}[Y|\boldsymbol{X}_S, X_j]|\boldsymbol{X}_S)]$ (Covert et al., 2020). For both expressions, one can easily see that they evaluate to zero when X_j is conditionally independent of Y given $\boldsymbol{X}_S$, i.e. $X_j \perp Y \mid \boldsymbol{X}_S$.

As a consequence of Theorem 1, knowledge of the dependence structure in the data allows speeding up the SAGE estimation procedure: evaluations of $\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S)$ can be skipped if $X_j \perp Y \mid \mathbf{X}_S$.

To identify the CIs in the data, we suggest leveraging greedy procedures that were originally developed to learn

[4] The Bayesian Dirichlet equivalent uniform (BDeu) score satisfies the properties too and is a valid alternative.

Algorithm 1: Sampling-based Approximation of d-SAGE

```
Input: Data $\{\mathbf{x}_i, y_i\}_{i=1}^n$ with $\mathbf{x}_i \in \mathbb{R}^d$, model $\hat{f}$, loss
  function $\ell$, number of permutations $n_\pi$
Infer DAG $\mathcal{G}$ from data $\{\mathbf{x}_i, y_i\}_{i=1}^n$ with structure
  learning algorithm of choice.
for i in $\{1, ..., n_\pi\}$ do
    Sample a permutation $\pi$
    $S = \emptyset$
    for j in $\{1, ..., d\}$ do
        if $X_{\pi_j} \not\perp_{\mathcal{G}} Y | \boldsymbol{X}_S$ then
            Sample $\mathbf{x}_{\bar{S}}$ from $p(\mathbf{x}_{\bar{S}}|\mathbf{x}_S)$
            Sample $\mathbf{x}_{\overline{S\cup\pi_j}}$ from $p(\mathbf{x}_{\overline{S\cup\pi_j}}|\mathbf{x}_S)$, where
              $\pi_j$ is the $j$-th element of $\pi$
            $\hat{\Delta}_{j|S} = \ell(\hat{f}(\mathbf{x}_S, \mathbf{x}_{\bar{S}})) - \ell(\hat{f}(\mathbf{x}_{S\cup\pi_j}, \mathbf{x}_{\overline{S\cup\pi_j}}))$
        else
            $\hat{\Delta}_{j|S} = 0$
        end
        $S = S \cup \pi_j$
    end
end
return $\hat{\phi}_j = \frac{1}{n_\pi} \sum_{i=1}^{n_\pi} \hat{\Delta}_{j|S}$  for $j = 1, ..., d$
```

Note that we dropped indices of $\hat{\Delta}_{j|S}$ for readability.

the causal structure in the data. CSL algorithms allow the estimation of a causal graph in polynomial time (Scutari et al., 2019b). Given that the Markov property and faithfulness are fulfilled, the graph allows reading off (conditional) independencies in the data using linear time d-separation queries (Hagberg et al., 2008; Darwiche, 2009). Our rationale is that the one-time effort of learning the causal graph, as well as the additional linear time d-separation queries, are negligible in comparison to the computational overhead of computing the surplus contributions.[5]

To summarise, d-SAGE estimation introduces two key differences to the original SAGE approximation algorithm. First, a graph $\mathcal{G}$ is fitted over all random variables, the features, and the target. Second, the estimation of $\Delta_{j|S}$ is skipped if the current feature X_j in permutation π is d-separated from the target given the set $\mathbf{X}_S = \{X_i : \pi(i) < \pi(j)\}$. The changes are highlighted in blue in Algorithm 1.

5 EXPERIMENTS

The experiment section is divided into three parts. In the first two parts, we evaluate our method on synthetic data with known ground truth: As we use d-separation queries in estimated graphs for d-SAGE approximation, we first evaluate the accuracy of d-separations in learned structures with regard to ground truth CIs in the data (Section 5.1). Then we compare d-SAGE to ordinary SAGE value approximation (Section 5.2). In the third part, we demonstrate the usefulness of the method in a real-world application (Section 5.3).[6]

5.1 Benchmark of Causal Structure Learning

Existing structure learning benchmarks evaluate the algorithms regarding how well they can recover the true causal structure (Constantinou et al., 2021; Scutari et al., 2019a). For d-SAGE, however, we are only interested in learning the dependence structure. As such, we assess how well CIs are inferred as d-separations in the estimated graph.

5.1.1 Setup

We evaluate the greedy search algorithms HC and TABU (Scutari et al., 2019b; Russell and Norvig, 2009). We selected these methods based on their superior performance in recent CSL benchmarks (Constantinou et al., 2021; Scutari et al., 2019a). As performance metrics, we employ the F1 score for the detection of d-separations w.r.t. a randomly sampled target Y as well as the respective false discovery rate. More precisely, for every potential d-separation of the form $X_j \perp_{\mathcal{G}} Y|\mathbf{X}_S$, we check whether it had the same status in the ground truth and the estimated graph. To cope with the exponentially large number of d-separations in the higher dimensional graphs (DAG$_{sm}$, DAG$_m$ and DAG$_l$) we randomly sampled a node of interest X_j and a conditioning set $\mathbf{X}_S$ one million times instead of iterating over all potential d-separation statements. For both algorithms, we relied on their implementation in *bnlearn* (Scutari, 2010) for R.[7] We consider twelve different synthetic data settings with known ground truth:

DAG$_s$, DAG$_{sm}$, DAG$_m$ and DAG$_l$ We sampled synthetic graphs with a varying number of nodes ($s = 10$, $sm = 20$, $m = 50$ and $l = 100$) and three different densities (average adjacency degrees of 2, 3 and 4). Based on the graphs, we sampled data from the corresponding linear Gaussian data model, where absolute values of edge weights are bounded by 0.5 and 2. We standardised variances to be (approximately) one to avoid that they increase with the topological ordering and counteract a potential bias in the benchmark (Reisach et al., 2021). For the sampling itself, we relied on the the *pcalg* package (Kalisch et al., 2012) implemented in R (R Core Team, 2022).

[5] In general, the complexity of conditional sampling depends on the assumptions about the data generating process. In their tractability analysis for SHAP, Van den Broeck et al. (2022) assume polynomial complexity for computing the conditional expectations of the form $\mathbb{E}_{\mathbf{X}_{\bar{S}}|\mathbf{X}_S}[\hat{f}(\mathbf{X})|\mathbf{X}_S = \mathbf{x}_S]$.

[6] All code is publicly available `https://github.com/gcskoenig/csl-experiments/tree/camera-ready`.

[7] All graph learning experiments were run on an Intel Core i7-8700K Desktop CPU.

5.1.2 Results

First, we observe TABU, while approximately taking double the time, either performs equally well as or better than HC (cf. Figures 1, 2 and Appendix C). Hence, we restrict this section to results for TABU search, which we also employed for d-SAGE estimation. Figure 1 shows the runtime of graph learning depending on sample size and corresponding F1 scores for d-separation inference for all twelve graphs. The key takeaway is that for the sparsest graph (average adjacency degree 2) the F1 score is greater than 0.88 if $n \geq 10,000$. For the larger graphs, however, there is a slight drop-off in performance, which is expected. Only for the densest graph setting (average adjacency degree 4) and for 50 and 100 nodes, though, a larger sample size, i.e. $n \geq 100,000$, is required to infer d-separations at a reasonable rate. As we will see in Section 5.2, the runtime for graph learning is negligible in the context of d-SAGE estimation.

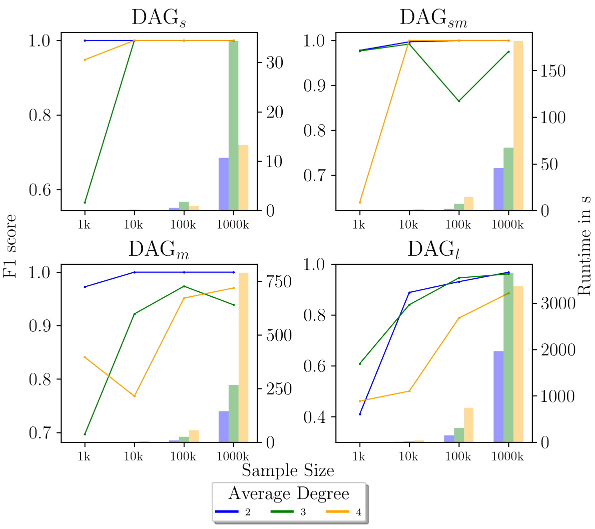


Figure 1: F1 scores for d-separation (lines, left y-axes) and runtime of graph learning (bars, right y-axes) using TABU search depending on sample size.

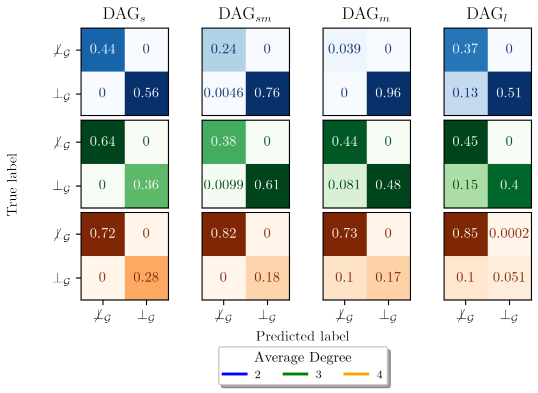


Figure 2: Confusion matrix for true and predicted d-connections ($\not\perp_{\mathcal{G}}$) and d-separations ($\perp_{\mathcal{G}}$) based on TABU search with $n = 10,000$ for all twelve graphs.

We note that there is no well-defined threshold for the minimal F1 score that would be required for SAGE estimation to benefit from causal structure learning because different error types have distinct consequences. While incorrectly inferred d-separations may lead to biased estimates, non-detected d-separations only reduce the benefit of CSL in terms of reduced runtime. Importantly, our simulation results in Figure 2 show virtually no false discoveries (cases where there is no CI in the underlying distribution but a d-separation is inferred) yet some false-negative instances, which leads to fewer skipped evaluations of $\Delta_{j|S}$ than warranted. This result is in accordance with the reasoning presented in Section 3.3. As such, the use of CSL is a conservative approach to the inference of CIs. Note that for the data used in the benchmark, the ground-truth graph is known and the Markov property and faithfulness hold, such that d-separations indeed coincide with statistical independence.

5.2 Evaluating Efficiency and Accuracy of d-SAGE

In the benchmark study in Section 5.1 we highlight the capability of structure learning to efficiently yet conservatively estimate d-separations as equivalents to CIs. We now evaluate d-SAGE regarding its efficiency and its accuracy.

5.2.1 Setup

To evaluate d-SAGE in practice, a linear model (LM) and a random forest (RF) are fitted to each of the twelve datasets (using the *scikit-learn* implementation with default settings (Pedregosa et al., 2011)). As loss function, the mean squared error (MSE) is used for either of them. Hence, the linear model (LM) falls into the category of optimal models required for the theoretical justification. The RF model serves as a sanity check for a high-performing, but not optimal model (cf. Appendix D for the model performances). For a fair comparison, we compare d-SAGE and SAGE based on the exact same feature orderings. This also allowed us to compare the skipped evaluations of $\Delta_{j|S}$, that are set to zero, to their estimated counterparts that should be very close to zero. Overall, we estimated SAGE and d-SAGE values five times for each setup (graph + model). We used the same synthetic datasets for the evaluation as in Section 5.1.

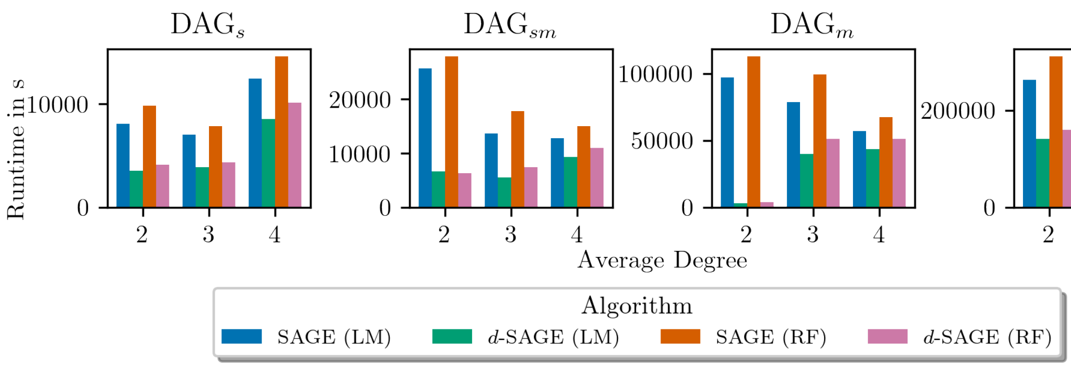
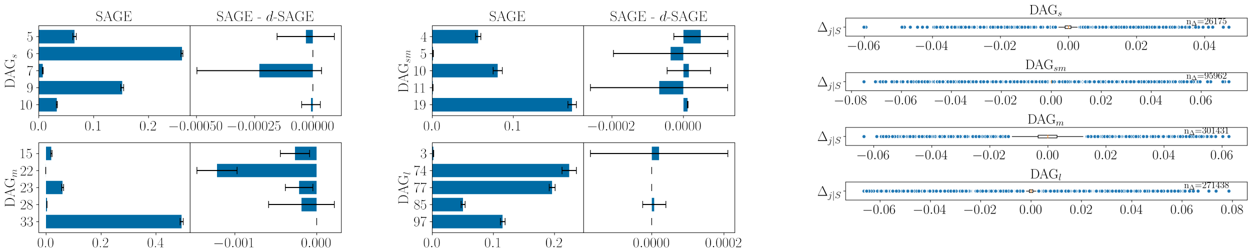


Figure 3: Runtime estimates for SAGE and d-SAGE for all twelve graphs and linear models (LM) as well as random forests (RF) based on $n = 10,000$.



(a) SAGE values and difference between SAGE and d-SAGE for the five largest values.

(b) Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

Figure 4: Results on the approximation quality of d-SAGE based on DAGs with average degree 2 for optimal models (LM). Based on five (d-)SAGE estimates.

5.2.2 Results

We find that d-SAGE indeed speeds up SAGE approximation as expected. More specifically, the estimated runtime[8] decreases by a rate that is approximately equal to the share of CIs w.r.t. the model target (cf. Appendix C) for both model classes across all graphs (cf. Figure 3). Furthermore, d-SAGE manages this speedup without distorting the estimates. Note that we do not include graph learning runtime in Figure 3 since it required between 0.06 seconds (DAG$_s$ with average degree 2) and 39.86 seconds (DAG$_l$ with average degree 4) and hence is negligible in this context.

Linear Model Figure 4 (a) displays the five SAGE values with the largest absolute value for the four graphs with an average degree of two along with the respective difference between the SAGE and d-SAGE estimates. Overall, the differences are about three orders of magnitude smaller than the original SAGE values, i.e. typically lie beneath one per cent. Even the most pronounced difference for variable 7 in DAG$_s$ only amounts to approximately 2.7 per cent of the SAGE value of approximately 0.007. We find no further striking differences in the remaining SAGE values that identified important features, i.e. those with the largest absolute SAGE values. Features deemed unimportant by SAGE values are detected as such by d-SAGE. Noteworthy, some d-SAGE estimates are equal to zero if the feature of interest is conditionally independent of the target given all (sampled) coalitions. Here, we argue that we bias observational SAGE values towards zero, which for truly independent features is closer (or equal) to the 'true-to-the-data' measure that would be achieved for the optimal predictor and infinite data.

Figure 4 (b) displays every $\Delta_{j|S}$ value, which was derived from a conditionally independent feature that was detected as such and thus set equal to zero in d-SAGE approximation. We see clearly that most values are very close to zero, as mirrored by the narrow boxes, which underlines the use-

[8] The complete SAGE estimation was performed on multiple different machines. For a fair evaluation of runtime, we relied on estimates that were performed on the same CPU (Intel Core i7-8700K Desktop CPU): Either approach was conducted using 100 permutations that were the same for SAGE and d-SAGE and runtime multiplied by the factor $\frac{n_\pi}{100}$, where n_π is the number of permutations after which one SAGE run converged. For convergence behaviour see Appendix E.

fulness of our approach.

Random Forest In order to test the sensitivity of the results, we replicated the exact same study using a high-performing but not optimal RF regressor (instead of the optimal LM). While the runtime savings are the same as for the LM, deviations of d-SAGE values from the original estimates are slightly more pronounced (cf. Appendix D). The results indicate that our approach is also useful for close to optimal models.

5.3 Real-world Application

To show the usefulness of d-SAGE in practice, we applied the approach to drug consumption data from the UCI ML repository (Dua and Graff, 2017). The target "Nicotine consumption" was predicted using logistic regression relying on twelve explanatory variables in a dataset with sample size $n = 1885$. Graph fitting was conducted with the TABU search algorithm and took 0.035 seconds. SAGE estimation for five different runs took approximately 12h14min[9]. To derive d-SAGE values, we did not rerun the estimation relying on d-SAGE but simply replaced the respective $\Delta_{j|S}$ that pertained to a d-separation in the fitted graph in the output (that included all such $\Delta_{j|S}$) with zero. We found approximately 38 per cent such $\Delta_{j|S}$ values that can be skipped which warrants an (almost) equally large relative speedup.

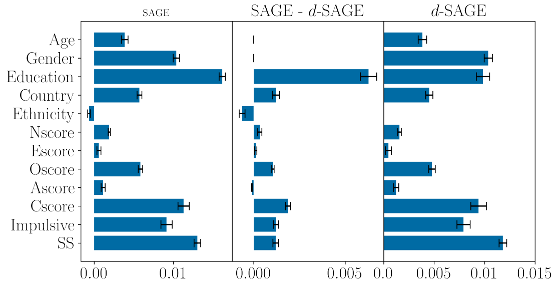


Figure 5: SAGE values, difference between SAGE and d-SAGE and d-SAGE values for drug consumption data. Based on five (d-)SAGE estimates.

Figure 5 shows that d-SAGE values are mostly in accordance with the original SAGE estimate. From the important variables, only *'Education'* has a markedly distinct d-SAGE value as it is reduced by about a third compared to the SAGE estimate. Yet, it is still assigned relatively high importance. The efficacy of d-SAGE in practice is further highlighted by the $\Delta_{j|S}$ values that hover around zero, as shown in Figure 6.

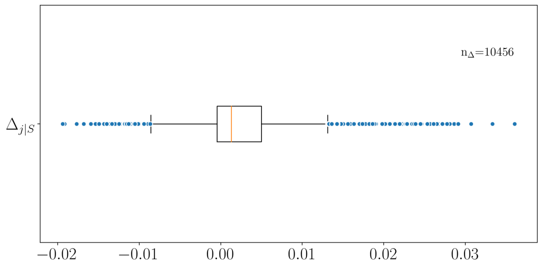


Figure 6: Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

6 DISCUSSION

Model optimality and loss Conditional SAGE values are particularly appealing for scientific inference, i.e. to learn about the data (Chen et al., 2020; Covert et al., 2020). Therefore, in general, accurate predictors are required (Molnar et al., 2022). However, the requirement is of increased importance for d-SAGE since if the assumption of model optimality is violated the interpretation may be further biased by skipping the evaluation of non-zero surplus contributions (Theorem 1).

Assumptions for CSL CSL is enabled by causal sufficiency, the Markov property and faithfulness (Peters et al., 2017). The assumptions ensure that all relevant variables are observed, and that CIs in the data coincide with d-separations in the true causal graph (which we assume to be a DAG). We conjecture that violations of these assumptions are not vital for our approach since learning the true causal graph is not the goal. Instead, we are only interested in learning the graph to encode (conditional) independencies present in the observational distribution (irrespective of which causal mechanism they stem from). DAGs learned by HC being a minimal I-Map of the underlying distribution makes it suitable for probabilistic inference of CIs without guarantees of a correct graph or the number of CIs uncovered.

Nevertheless, practitioners should carefully assess the assumptions before applying d-SAGE. In the presence of latent confounders or cyclic assignment, for example, one may consider other concepts, such as m-separation and σ-separation (cf. Bongers et al. (2021)). Moreover, it is advisable to perform sanity checks on whether skipped surplus contributions are actually evaluated to zero.

Use of Score-based CSL The analysis was restricted to the use of score-based CSL because of its efficiency. HC is particularly appealing since it infers a minimal I-Map of the underlying distribution as explained in Section 3.3, and TABU performed well empirically. However, inference of CIs is not limited to those techniques. Graph learning can be performed with an algorithm of choice and under

[9] All calculations were run on an Intel Core i7-8700K Desktop CPU

consideration of the assumptions employed, as explained above. Moreover, the rationale behind our approach is to replace CI testing by CSL. Partial correlation tests, for example, are considerably less efficient than d-separation queries (cf. Appendix F) and thus would require a larger number of CIs to achieve a speedup of SAGE.

7 CONCLUSION

We proposed d-SAGE, a method that exploits the dependence structure in the data to speed up SAGE estimation. More specifically, we observe that conditional independence in the data implies that the corresponding surplus contribution can be directly evaluated to zero. We modify the ordering based SAGE approximation algorithm to first learn the dependence structure in the data using CSL algorithms and to then skip surplus contribution evaluations if the graph encodes a CI. Errors in the learned graph may either slow down convergence (if CIs are not discovered) or bias the result towards zero (in case of false discoveries). However, in our experiments, there were nearly no false discoveries, such that the resulting estimates for features that were not conditionally independent given every coalition essentially converged to the same values as the original SAGE approximation algorithm. Furthermore, the CSL algorithms were able to uncover most CIs, such that we observe significant performance gains. As such, given a fixed computational budget, the efficiency gains of d-SAGE can enable a more accurate estimation of SAGE values than the approximation algorithm proposed by Covert et al. (2020). In future work, it would be interesting to combine d-SAGE with the permutation sampling by Mitchell et al. (2022) and to assess whether the results can be translated to other Shapley based interpretability methods such as SHAP.

Acknowledgements

This project is supported by the German Federal Ministry of Education and Research (BMBF). The computational results presented have been achieved in part using the Vienna Scientific Cluster (VSC). We thank Bernd Bischl for his advice on benchmark design.

References

Aas, K., Jullum, M., and Løland, A. (2021). Explaining individual predictions when features are dependent: More accurate approximations to shapley values. *Artificial Intelligence*, 298:103502.

Bashtannyk, D. M. and Hyndman, R. J. (2001). Bandwidth selection for kernel conditional density estimation. *Computational Statistics & Data Analysis*, 36(3):279–298.

Bénard, C., Biau, G., Da Veiga, S., and Scornet, E. (2022). Shaff: Fast and consistent shapley effect estimates via random forests. In *International Conference on Artificial Intelligence and Statistics*, pages 5563–5582. PMLR.

Bhatt, U., Xiang, A., Sharma, S., Weller, A., Taly, A., Jia, Y., Ghosh, J., Puri, R., Moura, J. M., and Eckersley, P. (2020). Explainable machine learning in deployment. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 648–657.

Bishop, C. M. (1994). Mixture density networks. Technical report, Aston University.

Blesch, K., Watson, D. S., and Wright, M. N. (2022). Conditional feature importance for mixed data. *arXiv preprint arXiv:2210.03047*.

Bongers, S., Forré, P., Peters, J., and Mooij, J. M. (2021). Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49(5):2885 – 2915.

Chen, H., Janizek, J. D., Lundberg, S., and Lee, S.-I. (2020). True to the model or true to the data? *arXiv preprint arXiv:2006.16234*.

Chickering, D. M. (2003). Optimal structure identification with greedy search. *J. Mach. Learn. Res.*, 3(null):507–554.

Constantinou, A. C., Liu, Y., Chobtham, K., Guo, Z., and Kitson, N. K. (2021). Large-scale empirical validation of bayesian network structure learning algorithms with noisy data. *International Journal of Approximate Reasoning*, 131:151–188.

Covert, I. and Lee, S.-I. (2021). Improving kernelshap: Practical shapley value estimation using linear regression. In *International Conference on Artificial Intelligence and Statistics*, pages 3457–3465. PMLR.

Covert, I., Lundberg, S. M., and Lee, S.-I. (2020). Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33:17212–17223.

Darwiche, A. (2009). *Modeling and Reasoning with Bayesian Networks*. Cambridge University Press.

Datta, A., Sen, S., and Zick, Y. (2016). Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE symposium on security and privacy (SP)*, pages 598–617. IEEE.

Dua, D. and Graff, C. (2017). UCI machine learning repository.

Farrell, S., Calafiura, P., Mudigonda, M., Anderson, D., Vlimant, J.-R., Zheng, S., Bendavid, J., Spiropulu, M., Cerati, G., Gray, L., et al. (2018). Novel deep learning methods for track reconstruction. *arXiv preprint arXiv:1810.06111*.

Freiesleben, T., König, G., Molnar, C., and Tejero-Cantero, A. (2022). Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena. *arXiv preprint arXiv:2206.05487*.

Frye, C., de Mijolla, D., Begley, T., Cowton, L., Stanley, M., and Feige, I. (2020a). Shapley explainability on the data manifold.

Frye, C., Rowat, C., and Feige, I. (2020b). Asymmetric shapley values: Incorporating causal knowledge into model-agnostic explainability. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA. Curran Associates Inc.

Gámez, J., Mateo, J. L., and Puerta, J. (2011). Learning bayesian networks by hill climbing: efficient methods based on progressive restriction of the neighborhood. *Data Mining and Knowledge Discovery*, 22(1):106–148.

Hagberg, A., Swart, P., and S Chult, D. (2008). Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM (United States).

Heskes, T., Sijben, E., Bucur, I. G., and Claassen, T. (2020). Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models. *Advances in neural information processing systems*, 33:4778–4789.

Hothorn, T. and Zeileis, A. (2021). Predictive distribution modeling using transformation forests. *Journal of Computational and Graphical Statistics*, 30(4):1181–1196.

Jethani, N., Sudarshan, M., Covert, I. C., Lee, S.-I., and Ranganath, R. (2021). Fastshap: Real-time shapley value estimation. In *International Conference on Learning Representations*.

Kalisch, M., Mächler, M., Colombo, D., Maathuis, M. H., and Bühlmann, P. (2012). Causal inference using graphical models with the R package pcalg. *Journal of Statistical Software*, 47(11):1–26.

Li, X., Zhou, Y., Dvornek, N. C., Gu, Y., Ventola, P., and Duncan, J. S. (2020). Efficient shapley explanation for features importance estimation under uncertainty. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 792–801. Springer.

Luan, H. and Tsai, C.-C. (2021). A review of using machine learning approaches for precision education. *Educational Technology & Society*, 24(1):250–266.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Mitchell, R., Cooper, J., Frank, E., and Holmes, G. (2022). Sampling permutations for shapley value estimation.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com.

Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C. A., Casalicchio, G., Grosse-Wentrup, M., and Bischl, B. (2022). General pitfalls of model-agnostic interpretation methods for machine learning models. In *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*, pages 39–68. Springer.

Obermeyer, Z. and Mullainathan, S. (2019). Dissecting racial bias in an algorithm that guides health decisions for 70 million people. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 89–89.

Page Jr, T. J. (1984). Multivariate statistics: A vector space approach. *JMR, Journal of Marketing Research (pre-1986)*, 21(000002):236.

Pearl, J. (2009). *Causality*. Cambridge University Press, 2 edition.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Peters, J., Janzing, D., and Schölkopf, B. (2017). *Elements of Causal Inference: Foundations and Learning Algorithms*. MIT Press, Cambridge, MA, USA.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Raghavan, M., Barocas, S., Kleinberg, J., and Levy, K. (2020). Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 469–481, New York, NY, USA. Association for Computing Machinery.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204.

Reisach, A. G., Seiler, C., and Weichwald, S. (2021). Beware of the simulated dag! causal discovery benchmarks may be easy to game.

Russell, S. and Norvig, P. (2009). *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edition.

Schmidt, J., Marques, M. R., Botti, S., and Marques, M. A. (2019). Recent advances and applications of machine learning in solid-state materials science. *npj Computational Materials*, 5(1):1–36.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464.

Scutari, M. (2010). Learning bayesian networks with the bnlearn r package. *Journal of Statistical Software*, 35(3):1–22.

Scutari, M., Graafland, C. E., and Gutiérrez, J. M. (2019a). Who learns better bayesian network structures: Accuracy and speed of structure learning algorithms. *International Journal of Approximate Reasoning*, 115:235–253.

Scutari, M., Vitolo, C., and Tucker, A. (2019b). Learning bayesian networks from big data with greedy search: computational complexity and efficient implementation. *Statistics and Computing*, 29(5):1095–1108.

Senior, A. W., Evans, R., Jumper, J., Kirkpatrick, J., Sifre, L., Green, T., Qin, C., Žídek, A., Nelson, A. W., Bridgland, A., et al. (2020). Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710.

Shapley, L. S. (1953). Stochastic games. *Proceedings of the national academy of sciences*, 39(10):1095–1100.

Sohn, K., Lee, H., and Yan, X. (2015). Learning structured output representation using deep conditional generative models. *Advances in neural information processing systems*, 28.

Spirtes, P., Glymour, C., and Scheines, R. (2000). *Causation, Prediction, and Search.* MIT press, 2nd edition.

Štrumbelj, E. and Kononenko, I. (2014). Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665.

Sundararajan, M. and Najmi, A. (2020). The many shapley values for model explanation. In *International conference on machine learning*, pages 9269–9278. PMLR.

Trippe, B. L. and Turner, R. E. (2018). Conditional density estimation with bayesian normalising flows. *arXiv preprint arXiv:1802.04908*.

Vallat, R. (2018). Pingouin: statistics in python. *Journal of Open Source Software*, 3(31):1026.

Van den Broeck, G., Lykov, A., Schleich, M., and Suciu, D. (2022). On the tractability of shap explanations. *Journal of Artificial Intelligence Research*, 74:851–886.

Wang, J., Wiens, J., and Lundberg, S. (2021). Shapley flow: A graph-based approach to interpreting model predictions. In *International Conference on Artificial Intelligence and Statistics*, pages 721–729. PMLR.

Winkler, C., Worrall, D., Hoogeboom, E., and Welling, M. (2019). Learning likelihoods with conditional normalizing flows. *arXiv preprint arXiv:1912.00042*.

Zeng, J., Ustun, B., and Rudin, C. (2017). Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 180(3):689–722.

2.7 Paper VII: Model-agnostic Feature Importance and Effects with Dependent Features

Molnar, Christoph, Gunnar König, Bernd Bischl, Giuseppe Casalicchio. **Model-agnostic Feature Importance and Effects with Dependent Features: A Conditional Subgroup Approach.** *Data Mining and Knowledge Discovery (2023):* 1-39.

Gunnar König contributed to the paper as co-author with significant contributions. Gunnar Koenig wrote large parts of Section 1 and Section 4 and provided the proofs in Appendix A and B. Christoph Molnar wrote most of the paper. All authors added input, suggested modifications proofread and revised the paper.

Data Mining and Knowledge Discovery
https://doi.org/10.1007/s10618-022-00901-9

Model-agnostic feature importance and effects with dependent features: a conditional subgroup approach

Christoph Molnar[1,4] · Gunnar König[1,2,3] · Bernd Bischl[1,3] · Giuseppe Casalicchio[1,3]

Received: 14 June 2021 / Accepted: 19 November 2022

Abstract

The interpretation of feature importance in machine learning models is challenging when features are dependent. Permutation feature importance (PFI) ignores such dependencies, which can cause misleading interpretations due to extrapolation. A possible remedy is more advanced conditional PFI approaches that enable the assessment of feature importance conditional on all other features. Due to this shift in perspective and in order to enable correct interpretations, it is beneficial if the conditioning is transparent and comprehensible. In this paper, we propose a new sampling mechanism for the conditional distribution based on permutations in conditional subgroups. As these subgroups are constructed using tree-based methods such as transformation trees, the conditioning becomes inherently interpretable. This not only provides a simple and effective estimator of conditional PFI, but also local PFI estimates within the subgroups. In addition, we apply the conditional subgroups approach to partial dependence plots, a popular method for describing feature effects that can also suffer from extrapolation when features are dependent and interactions are present in the model. In simulations and a real-world application, we demonstrate the advantages of the conditional subgroup approach over existing methods: It allows to compute conditional PFI that is more true to the data than existing proposals and enables a fine-grained interpretation of feature effects and importance within the conditional subgroups.

Responsible editor: Martin Atzmueller, Johannes Fürnkranz, Tomáš Kliegr and Ute Schmid

Giuseppe Casalicchio
giuseppe.casalicchio@stat.uni-muenchen.de

[1] Department of Statistics, Ludwig-Maximilians-University Munich, Munich, Germany

[2] Research Group Neuroinformatics, University of Vienna, Vienna, Austria

[3] Munich Center for Machine Learning (MCML), Munich, Germany

[4] Leibniz Institute for Prevention Research and Epidemiology - BIPS GmbH, Bremen, Germany

Published online: 10 January 2023

Keywords Interpretable machine learning · Explainable AI · Permutation feature importance · Partial dependence plot

1 Introduction

A promising avenue of research suggests to make inference about the data generating process by analyzing machine learning models using Interpretable Machine Learning (IML). The Partial Dependence Plot (PDP) (Friedman et al. 1991) and Permutation Feature Importance (PFI) (Breiman 2001) are model-agnostic tools (working for all kinds of machine learning models) that have been used for scientific discoveries. Applications range from medicine (Boulesteix et al. 2020; Stiglic et al. 2020; Pintelas et al. 2020) and the social sciences (Stachl et al. 2020; Zhao et al. 2020) to ecology (Bair et al. 2013; Esselman et al. 2015; Obringer and Nateghi 2018). PDP and PFI are used to study effect and importance of features: The PDP visualizes how a change in a feature, on average, changes the predicted outcome; the PFI ranks the features based on how much they contribute to the model performance.

Both PDP and PFI rely on marginal sampling of feature values. A range of work argues that marginal-sampling based interpretation techniques, including PDP and PFI, are not suitable for learning about the data generating process (Hooker and Mentch 2019; Frye et al. 2020; Chen et al. 2020; Freiesleben et al. 2022). The reason is that marginal-sampling based techniques ignore dependencies between the features and as a consequence may explain the model's behaviour in unlikely or even unrealistic regions of the feature space.

As a solution, conditional-sampling based techniques, such as conditional permutation feature importance (cPFI) and conditional partial dependence plots (cPDP) were proposed which only evaluate the model within the joint distribution (Strobl et al. 2008; Apley and Zhu 2016; Hooker and Mentch 2019). Given loss-optimal models, they allow insights into the data generating process. More specifically, cPFI allows to quantify whether knowing a feature is required to achieve the same predictive performance, such that nonzero cPFI can be linked with conditional dependence in the data (König et al. 2020). cPDPs visualize the relationships in the data (through the model's perspective), i.e. they describe how the conditional expectation of the outcome varies with the feature of interest (Freiesleben et al. 2022).

Although theoretically appealing, conditional-sampling based methods are more difficult to apply than marginal-sampling based methods. Existing proposals for cPFI require sampling from the conditional distribution of the feature of interest given the remaining features, which is challenging. The estimation of cPDP is especially challenging, since sampling from the multivariate conditional of the remaining features given the feature of interest is required.

Contributions: Instead of modeling the conditional distribution, we suggest to learn a tree-based partitionioning of the feature space into blocks within which the feature of interest is not (or at least less) correlated with the remaining features. This partitioning can be leveraged in several ways to derive interpretations that allow interesting insight. First of all, we can compute the well-established global cPFI by computing the PFI for each subgroup and aggregating the result. Leveraging the flexibility of tree-based learners, this approach allows the computation of cPFI for mixed continuous and categorical data. Secondly, in situations where the partitioning requires only a few splits, the partitioning itself is interpretable. We can then leverage the partitioning to (a) get insight into the dependence structure in the data and (b) derive subgroup specific versions of PFI and PDP, to also understand under which circumstances variables are relevant or have a certain effect. For instance, by applying PFI in each subgroup, we find that temperature is not predictive of bike rentals given that we know it's summer, but highly predictive if we know that it's winter. Furthermore, by looking at the PDP within each subgroup we can understand how the conditional expectation varies with temperature given that we know that it's winter.

The paper is structured as follows: We introduce our notation in Sect. 2 and discuss related work in Sect. 3. We motivate and formally introduce the conditional subgroup approach in Sect. 4. We demonstrate the usefulness of the method on benchmarks with synthetic and real data (Sect. 5) and illustrate its interpretation in a real-world application (Sect. 6).

2 Notation and background

We consider ML prediction functions $\hat{f} : \mathbb{R}^p \mapsto \mathbb{R}$, where $\hat{f}(\boldsymbol{x})$ is a model prediction and $\boldsymbol{x} \in \mathbb{R}^p$ is a p-dimensional feature vector. We use $\mathbf{x}_j \in \mathbb{R}^n$ to refer to an observed feature (vector) and X_j to refer to the j-th feature as a random variable. With $\mathbf{x}_{-j}$ we refer to the complementary feature values $\mathbf{x}_{\{1,\ldots,p\}\setminus\{j\}} \in \mathbb{R}^{n\times(p-1)}$ and with X_{-j} to the corresponding random variables. We refer to the value of the j-th feature from the i-th instance as $x_j^{(i)}$ and to the tuples $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^n$ as data.

The *Permutation feature importance (PFI)* (Breiman 2001; Fisher et al. 2019) is defined as the increase in loss when feature X_j is permuted:

$$PFI_j = \mathbb{E}[L(Y, \hat{f}(\tilde{X}_j, X_{-j}))] - \mathbb{E}[L(Y, \hat{f}(X_j, X_{-j}))] \tag{1}$$

The theoretical PFI for a feature X_j is the difference between the expected loss when the feature is permuted and the original loss. If the random variable $\tilde{X}_j$ has the same marginal distribution as X_j (e.g., permutation), the estimate yields the marginal PFI. If $\tilde{X}_j$ follows the conditional distribution $\tilde{X}_j \sim X_j|X_{-j}$, we speak of the conditional PFI. The PFI is estimated with the following formula:

$$\widehat{PFI}_j = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{M}\sum_{m=1}^{M}\left(\tilde{L}_m^{(i)} - L^{(i)}\right)\right) \tag{2}$$

where $L^{(i)} = L(y^{(i)}, \hat{f}(\boldsymbol{x}^{(i)}))$ is the loss for the i-th observation and $\tilde{L}_m^{(i)} = L(y^{(i)}, \hat{f}(\tilde{x}_j^{(i)}, \boldsymbol{x}_{-j}^{(i)}))$ is the loss where $x_j^{(i)}$ was replaced by the m-th sample of $\tilde{x}_j^{(i)}$. The latter refers to the i-th feature value obtained by a sample of $\mathbf{x}_j$. The sample can be repeated M-times for a more stable estimation of $\tilde{L}^{(i)}$. Numerous variations of this formulation exist. Breiman (2001) proposed the PFI for random forests, which is computed from the out-of-bag samples of individual trees. Subsequently, Fisher et al. (2019) introduced a model-agnostic PFI version.

The marginal *partial dependence plot (PDP)* (Friedman et al. 1991) describes the average effect of the j-th feature on the prediction.

$$PDP_j(x) = \mathbb{E}[\hat{f}(x, X_{-j})] \tag{3}$$

The theoretical PDP is a marginalized version of the prediction function. All features with the exception of X_j are integrated out, and the p-dimensional prediction function becomes a 1-dimensional function, the PDP. There are two options: Integrate with respect to the marginal distribution $\P_{X_{-j}}$ or the conditional distribution $\P_{X_{-j}|X_j}$. If the expectation is conditional on X_j, $\mathbb{E}[\hat{f}(x, X_{-j})|X_j = x]$, we speak of the conditional PDP. The marginal PDP evaluated at feature value x is estimated using Monte Carlo integration.

$$\widehat{PDP}_j(x) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}\left(x, \boldsymbol{x}_{-j}^{(i)}\right) \tag{4}$$

In other words, at any given position x along the range of X_j, the PDP can be estimated by taking the data, setting $X_j = x$ for all observations and averaging the results.

3 Related work

In this section, we review conditional variants of PDP and PFI and other approaches that try to avoid extrapolation.

3.1 Related work on conditional PDP

The conditional PDP (M-Plot) (Apley and Zhu 2016) averages the predictions locally on the feature grid and mixes effects of dependent features. Apley and Zhu (2016) also address the interpretation problem that conditional PDP is influenced by feature effects of correlated features. The authors proposed accumulated local effect (ALE) plots, which reduce extrapolation by accumulating the finite differences computed within intervals of the feature of interest. By definition, interpretations of ALE plots are thus only valid locally within the intervals. Furthermore, there is no straightforward approach to derive ALE plots for categorical features, since ALE requires ordered feature values. Our proposed approach can handle categorical features.

Hooker (2007) proposed a functional ANOVA decomposition with hierarchically orthogonal components, based on integration using the joint distribution of the data, which in practice is difficult to estimate.

Another PDP variant based on stratification was proposed by Parr and Wilson (2019). However, this stratified PDP describes only the data and is independent of the model.

Individual conditional expectation (ICE) curves by Goldstein et al. (2015) can be used to visualize the interactions underlying a PDP, but they also suffer from the extrapolation problem. The "conditional" in ICE refers to conditioning on individual observations and not on certain features. As a solution, Hooker and Mentch (2019) suggested to visually highlight the areas of the ICE curves in which the feature combinations are more likely.

3.2 Related work on conditional PFI

We review approaches that modify the PFI (Breiman 2001; Fisher et al. 2019) in presence of dependent features by using a conditional sampling strategy.

Strobl et al. (2008) proposed the conditional variable importance for random forests (CVIRF), which is a conditional PFI variant of Breiman (2001). CVIRF was further analyzed and extended by Debeer and Strobl (2020). Both CVIRF and our approach rely on permutations based on partitions of decision trees. However, there are fundamental differences. CVIRF is specifically developed for random forests and relies on the splits of the underlying individual trees of the random forest for the conditional sampling. In contrast, our cs-PFI approach trains decision trees for each feature using X_{-j} as features and X_j as the target. Therefore, the subgroups for each feature are constructed from their conditional distributions (conditional on the other features) in a separate step, which is decoupled from the machine learning model to be interpreted. Our cs-PFI approach is model-agnostic, independent of the target to predict and not specific to random forests.

Hooker and Mentch (2019) made a general suggestion to replace feature values by estimates of $\mathbb{E}[X_j|X_{-j}]$.

Fisher et al. (2019) suggested to use matching and imputation techniques to generate samples from the conditional distribution. If X_{-j} has few unique combinations, they suggested to group $x_j^{(i)}$ by unique $\boldsymbol{x}_{-j}^{(i)}$ combinations and permute them for these fixed groups. For discrete and low-dimensional feature spaces, they suggest nonparametric matching and weighting methods to replace X_j values. For continuous or high-dimensional data, they suggest imputing X_j with $\mathbb{E}[X_j|X_{-j}]$ and adding residuals (under the assumption of homogeneous residuals). Our approach using permutation in subgroups can be seen as a model-driven, binary weighting approach extended to continuous features.

Knockoffs (Candes et al. 2018) are random variables which are "copies" of the original features that preserve the joint distribution but are independent of the prediction target conditional on the remaining features. Knockoffs can be used to replace feature values for conditional feature importance computation. Watson and Wright (2021) developed a testing framework for PFI based on knockoff samplers such as Model-X knockoffs (Candes et al. 2018). Our approach is complementary since Watson and Wright (2021) is agnostic to the sampling strategy that is used. Others have proposed to use generative adversarial networks for generating knockoffs (Romano et al. 2019). Knockoffs are not transparent with respect to how they condition on the features, while our approach creates interpretable subgroups.

Conditional importance approaches based on model retraining have been proposed (Hooker and Mentch 2019; Lei et al. 2018; Gregorutti et al. 2017). However, retraining the model can be expensive, and answers a fundamentally different question, often related to feature selection and not based on a fixed set of features. Hence, we focus on approaches that compute conditional PFI for a fixed model without retraining.

None of the existing approaches makes the dependence structures between the features explicit. It is unclear which of the features in X_{-j} influenced the replacement of X_j the most and how. Furthermore, little attention has been paid on evaluating how well different sampling strategies address the extrapolation problem. We address this gap with an extensive data fidelity experiment on the OpenML-CC18 benchmarking suite. To the best of our knowledge, our paper is also the first to conduct experiments using ground truth for the conditional PFI. Our approach works with any type of feature, be it categorical, numerical, ordinal and so on, since we rely on decision trees to find the subgroups used for conditioning. Further we are the first to discuss the trade-off between conditional and marginal PFI and PDP in depth. The differences between the different (conditional) PDP and PFI approaches ultimately boil down to how they sample from the conditional distribution. Table 1 lists different sampling strategies of model-agnostic interpretation methods and summarizes their assumptions to preserve the joint distribution.

4 Conditional subgroups

In this section, we propose a subgroup-based approach that allows us to (1) estimate the cPFI and to (2) introduce novel subgroup-specific versions of PDP and PFI that allow novel insight into model and data.

More specifically, we suggest to leverage tree-based learners to partition the feature space into groups G_j within which X_j is independent of the remaining features X_{-j} (Sect. 4.1). Permuting observations within such groups does not lead to extrapolation, because in each group the marginal and the conditional distribution coincide. We illustrate the idea in Fig. 1.

As a consequence, we can compute the cPFI by applying the PFI in each subgroup and aggregating the results (Sect. 4.2). Furthermore, if the data allow for a human-intelligible partitioning, we can also interpret the subgroup-wise PFI and PDP to gain novel insight about the circumstances given which variables are relevant or have a certain effect on the prediction (Sects. 4.2 and 4.3).

4.1 Learning conditional subgroups

In order to learn the grouping G_j, any algorithm can be used that splits the data in X_{-j} so that the distribution of X_j becomes more homogeneous within a group and more heterogeneous between groups. We consider decision tree algorithms for this task, which predict X_j based on splits in X_{-j}. Decision tree algorithms directly or indirectly optimize splits for heterogeneity of some aspects of the distribution of X_j in the splits. The partitions in a decision tree can be described by decision rules that lead to

Table 1 Sampling strategies for model-agnostic interpretation techniques

Sampling strategy	Used/suggested by	Assumptions
No intervention on X_j	Drop-and-Refit, LOCO (Lei et al. 2018)	
Permute X_j	Marginal PFI (Breiman 2001; Fisher et al. 2019), PDP (Friedman et al. 1991)	$X_j \perp\!\!\!\perp X_{-j}$
Replace X_j by knockoff Z_j with $(Z_j, X_{-j}) \sim (X_j, X_{-j})$ and $Z_j \perp\!\!\!\perp Y$	Knockoffs (Candes et al. 2018), CPI (Watson and Wright 2021)	$(X_j, X_{-j}) \sim N$
Move each $x_j^{(i)}$ to left and right interval bounds	ALE (Apley and Zhu 2016)	$X_j \perp\!\!\!\perp X_{-j}$ in intervals
Permute X_j in subgroups	cs-PFI, cs-PDP	$X_j \perp\!\!\!\perp X_{-j}$ in subgroups
Permute X_j in random forest tree nodes	CVIRF (Strobl et al. 2008; Debeer and Strobl 2020)	$X_j \perp\!\!\!\perp X_{-j}$ cond. on tree splits in X_{-j} to predict Y
Impute X_j from X_{-j}	(Fisher et al. 2019)	Homogeneous residuals

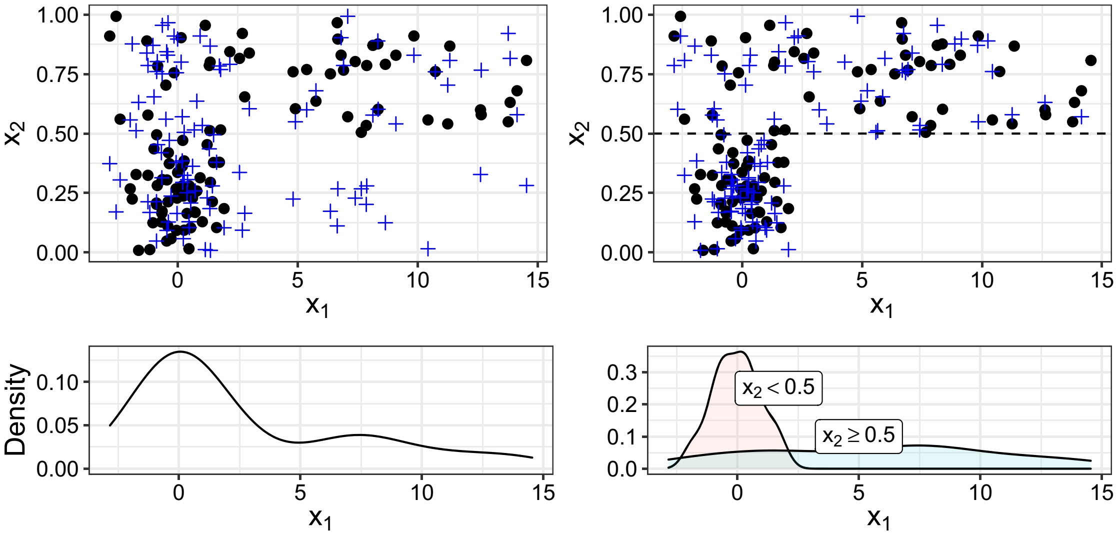


Fig. 1 Features $X_2 \sim U(0, 1)$ and $X_1 \sim N(0, 1)$, if $X_2 < 0.5$, else $X_1 \sim N(4, 4)$ (black dots). Top left: The crosses are permutations of X_1. For $X_2 < 0.5$, the permutation extrapolates. Bottom left: Marginal density of X_1. Top right: Permuting X_1 within subgroups based on X_2 ($X_2 < 0.5$ and $X_2 \geq 0.5$) reduces extrapolation. Bottom right: Densities of X_1 conditional on the subgroups

that terminal leaf. We leverage this partitioning to construct groups $\mathcal{G}_j^1, \ldots, \mathcal{G}_j^K$ based on random variable G_j for a specific feature X_j. The new variable can be calculated by assigning every observation the indicator of the partition that it lies in (meaning for observation i with $x_{-j}^{(i)} \in \mathcal{G}_j^k$ the group variable's value is defined as $g_j^{(i)} := k$).

Transformation trees (trtr) (Hothorn and Zeileis 2017) are able to model the conditional distribution of a variable. This approach partitions the feature space so that the distribution of the target (here X_j) within the resulting subgroups $\mathcal{G}_j^k$ is homogeneous, which means that the group-wise parameterization of the modeled distribution is independent of X_{-j}. Transformation trees directly model the target's distribution ¶$(X_j \leq x) = F_Z(h(x))$, where F_Z is the chosen (cumulative) distribution function and h a monotone increasing transformation function (hence the name transformation trees). The transformation function is defined as $\mathbf{a}(y)^T \boldsymbol{\theta}$ where $\mathbf{a} : \mathbb{R} \mapsto \mathbb{R}^k$ is a basis function of polynomials or splines. The task of estimating the distribution is reduced to estimating $\boldsymbol{\theta}$, and the trees are split based on hypothesis tests for differences in $\boldsymbol{\theta}$ given X_{-j}, and therefore differences in the distribution of X_j. For more detailed explanations of transformation trees please refer to Hothorn and Zeileis (2017).

In contrast, a simpler approach would be to use *classification and regression trees (CART)* (Breiman et al. 1984), which, for regression, minimizes the variance within nodes, effectively finding partitions with different means in the distribution of X_j. However, CART's split criterion only considers differences in the expectation of the distribution of X_j given X_{-j}: $\mathbb{E}[X_j|X_{-j}]$. This means CART could only make X_j and X_{-j} independent if the distribution of X_j only depends in its expectation on X_{-j} (and if the dependence can be modeled by partitioning the data). Any differences in higher moments of the distribution of X_j such as the variance of $X_j|X_{-j}$ cannot be detected.

We evaluated both trtr, which are theoretically well equipped for splitting distributions and CART, which are established and well-studied. For the remainder of this

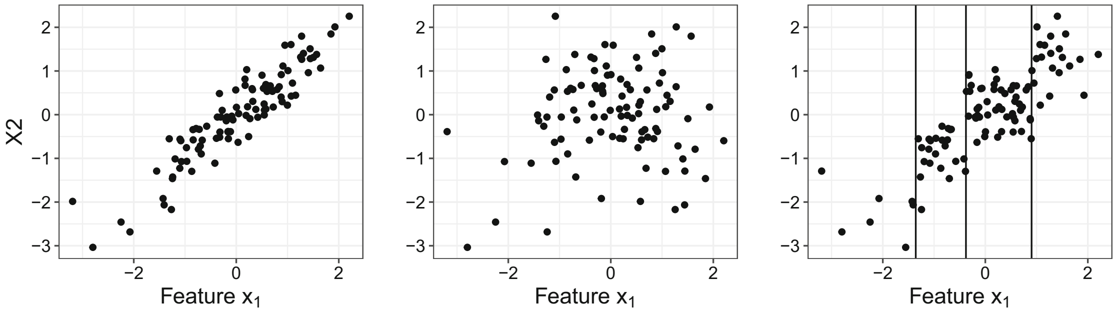


Fig. 2 Left: Simulation of features $X_1 \sim N(0, 1)$ and $X_2 \sim N(0, 1)$ with a covariance of 0.9. Middle: Unconditional permutation extrapolates strongly. Right: Permuting on partitions found by CART (predicting X_2 from X_1) has greatly reduced extrapolation, but cannot get rid of it completely. x_1 and x_2 remain correlated in the partitions

paper, we have set the default minimum number of observations in a node to 30 for both approaches. For the transformation trees, we used the Normal distribution as target distribution and we used Bernstein polynomials of degree five for the transformation function. Higher-order polynomials do not seem to increase model fit further (Hothorn 2018).

We denote the subgroups by $\mathcal{G}_j^k \subset \mathbb{R}^{p-1}$, where $k \in \{1, \ldots, K_j\}$ is the k-th subgroup for feature j, with K_j groups in total for the j-th feature. The subgroups per feature are disjoint: $\mathcal{G}_j^l \cap \mathcal{G}_j^k = \emptyset, \forall l \neq k$ and $\bigcup_{k=1}^{K} \mathcal{G}_j^k = \mathbb{R}^{p-1}$. Let $(\mathbf{y}_j^k, \mathbf{x}_j^k)$ be a subset of $(\mathbf{y}, \mathbf{x})$ that refers to the data subset belonging to the subgroup $\mathcal{G}_j^k$. Each subgroup can be described by the decision path that leads to the respective terminal node.

4.1.1 Remarks

Continuous dependencies For conditional independence $X_j \perp X_{-j} | G_j^k$ to hold, the chosen decision tree approach has to capture the (potentially complex) dependencies between X_j and X_{-j}. CART can only capture differences in the expected value of $X_j | X_{-j}$ but are insensitive to changes in, for example, the variance. Transformation trees are in principle agnostic to the specified distribution and the default transformation family of distributions is very general, as empirical results suggest (Hothorn and Zeileis 2017). However, the approach is based on the assumption that the dependence can be modeled with a discrete grouping. For example, in the case of linear Gaussian dependencies, the corresponding optimal variable would be linear Gaussian itself, and would be in conflict with our proposed interpretable grouping approach. Even in these settings the approach allows an approximation of the conditional distribution. In the case of simple linear Gaussian dependencies, partitioning the feature space will still *reduce extrapolation*. But we never get rid of it completely, unless there are only individual data points left in each partition, see Fig. 2.

Sparse subgroups Fewer subgroups are generally desirable for two reasons: (1) for a good approximation of the marginal distribution within a subgroup, a sufficient number of observations per group is required, which might lead to fewer subgroups, and (2) a large number of subgroups leads to more complex groups, which reduces their human-intelligibility and therefore forfeits the added value of the local, subgroup-wise interpretations. As we rely on decision trees, we can adjust the granularity of the grouping using hyperparameters such as the maximum tree depth. By controlling the maximum tree depth, we can control the trade-off between the depth of the tree (and hence its interpretability) and the homogeneity of the distribution within the subgroups.

4.2 Conditional subgroup permutation feature importance (cs-PFI)

We estimate the cs-PFI of feature X_j within a subgroup $\mathcal{G}_j^k$ as:

$$PFI_j^k = \frac{1}{n_k} \sum_{i:\mathbf{x}^{(i)} \in \mathcal{G}_j^k} \left(\frac{1}{M} \sum_{m=1}^{M} L\left(y^{(i)}, \hat{f}\left(\tilde{x}_{j,m}^{(i)}, \mathbf{x}_{-j}^{(i)}\right)\right) - L\left(y^{(i)}, \hat{f}\left(\mathbf{x}^{(i)}\right)\right)\right), \tag{5}$$

where $\tilde{x}_{j,m}^{(i)}$ refers to a feature value obtained from the m-th permutation of x_j within the subgroup k_j. This estimation is exactly the same as the marginal PFI [Eq. (2)], except that it only includes observations from the given subgroup. Algorithm 1 describes the estimation of the cs-PFIs for a given feature on unseen data.

```
Algorithm 1: Estimate cs-PFI
   Input: Model $f$; data $\mathcal{D}_{train}$, $\mathcal{D}_{test}$; loss $L$; feature $j$; no. permutations $M$
 1 Train tree $T_j$ with target $X_j$ and features $X_{-j}$ using $\mathcal{D}_{train}$
 2 Compute subgroups $\mathcal{G}_j^k$ for $\mathcal{D}_{test}$ based on terminal nodes of $T_j$, $k \in \{1, \dots, K_j\}$
 3 for $k \in \{1, \dots, K_j\}$ do
 4     $L_{orig} := \frac{1}{n_k} \sum_{i:\boldsymbol{x}^{(i)} \in \mathcal{G}_j^k} L(y^{(i)}, \hat{f}(\boldsymbol{x}^{(i)}))$
 5     for $m \in \{1, \dots, M\}$ do
 6         Generate $\tilde{\boldsymbol{x}}_j^m$ by permuting feature values $\boldsymbol{x}_j$ within subgroup $\mathcal{G}_j^k$
 7         $L_{perm}^m := \frac{1}{n_k} \sum_{i:\boldsymbol{x}^{(i)} \in \mathcal{G}_j^k} L(y^{(i)}, \hat{f}(\tilde{x}_{j,m}^{(i)}, \boldsymbol{x}_{-j}^{(i)}))$
 8     cs-PFI$_j^k$ = $\frac{1}{M} \sum_{m=1}^{M} L_{perm}^m - L_{orig}$
 9 cs-PFI$_j$ = $\frac{1}{n} \sum_{k=1}^{K_j} n_k PFI_j^k$
```

The algorithm has two outcomes: We get local importance values for feature X_j for each subgroup (cs-PFI$_j^k$; Algorithm 1, line 8) and a global conditional feature importance (cs-PFI$_j$; Algorithm 1, line 9). The latter is equivalent to the weighted average of subgroup importances regarding the number of observations within each subgroup (see proof in "Appendix Appendix A)".

$$\text{cs-PFI}_j = \frac{1}{n} \sum_{k=1}^{K_j} n_k PFI_j^k$$

The cs-PFIs needs the same amount of model evaluations as the PFI ($O(nM)$). On top of that comes the cost for training the respective decision trees and making predictions to assign a subgroup to each observation.

Theorem 1 *When feature X_j is independent of features X_{-j} for a given dataset $\mathcal{D}$, each cs-PFI$_j^k$ has the same expectation as the marginal PFI, and an n/n_k-times larger variance, where n and n_k are the number of observations in the data and the subgroup $\mathcal{G}_j^k$.*

The proof of Theorem 1 is shown in "Appendix Appendix B". Theorem 1 has the practical implication that even in the case of applying cs-PFI to an independent feature, we will retrieve the marginal PFI, and not introduce any problematic interpretations. Equivalence in expectation and higher variance under the independence of X_j and X_{-j} holds true even if the partitions $\mathcal{G}_j^k$ would be randomly chosen. Theorem 1 has further consequences regarding overfitting: Assuming a node has already reached independence between X_j and X_{-j}, then further splitting the tree based on noise will not change the expected cs-PFIs.

4.3 Conditional subgroup partial dependence plots (cs-PDPs)

A range of work argues that PDPs are not suitable for inference if features are dependent (Hooker and Mentch 2019; Freiesleben et al. 2022). Conditional PDPs have been suggested as an alternative, but they are difficult to estimate, since they require sampling from the multivariate conditional of the remaining feature $P(X_{-j}|X_j)$. For settings where a human-intelligible partitioning can be learned, we suggest an alternative that does not require to sample from $P(X_{-j}|X_j)$: Instead of computing the global cPDP, we suggest to compute the cs-PDP$_j^k$ for each subgroup $\mathcal{G}_j^k$ using the marginal PDP formula in Eq. (4).

$$\text{cs-PDP}_j^k(x) = \frac{1}{n_k} \sum_{i:\mathbf{x}^{(i)} \in \mathcal{G}_j^k} \hat{f}\left(x, \mathbf{x}_{-j}^{(i)}\right)$$

This results in multiple cs-PDPs per feature, which can be displayed together in the same plot as in Fig. 9. The cs-PDPs allow interesting insight into data and model. First of all, since they do not extrapolate, they allow interesting insight into the data: They describe how prediction and feature of interest covary within specific groups. Secondly, in contrast to the global cPDP, they allow interesting insight into the model: For the global cPDP even features that are not used by the model can have nonzero effects (as illustrated in Fig. 3). Our proposed cs-PDPs only show nonzero effects if the respective variable is causal for the prediction.

4.3.1 Plotting the cs-PDP

The cs-PDP can be plotted in the same way as the PDP, with the exception that we get mutiple effect curves instead of just one. For a more compact view, we propose to

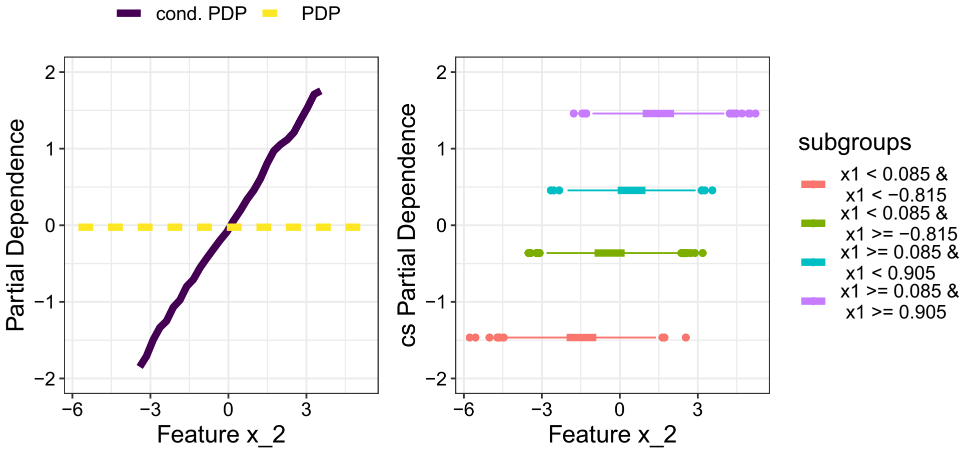


Fig. 3 We simulated a linear model of $y = x_1 + \epsilon$ with $\epsilon \sim N(0, 1)$ and an additional feature X_2 which is correlated with $X1$ (≈ 0.72). The conditional PDP (left) gives the false impression that X_2 has an influence on the target. The cs-PDPs help in this regard, as the effects due to X_1 (changes in intercept) are clearly separated from the effect that X_2 has on the target (slope of the cs-PDPs), which is zero. Unlike the marginal PDP, the cs-PDPs reveals that for increasing X_2 we expect that the prediction increases due to the correlation between X_1 and X_2

plot all cs-PDPs into the same plot. In addition, we suggest to plot the PDPs similar to boxplots, where the dense center quartiles are indicated with a bold line (see Fig. 4). By emphasizing the data density within the subgroups, the user can immediately see where to trust the plot more and where less. We restrict each cs-PDP$_j^k$ to the interval $[min(\boldsymbol{x}_j), max(\boldsymbol{x}_j)]$, with $\boldsymbol{x}_j = (x_j^{(1)}, \ldots, x_j^{(n_j^k)})$.

Equivalently to PFI, the subgroup PDPs approximate the true marginal PDP even if the features are independent.

Theorem 2 *When feature X_j is independent of features X_{-j} for a given dataset $\mathcal{D}$, each cs-PDP$_j^k$ has the same expectation as the marginal PDP, and an n/n_k-times larger variance, where n and n_k are the number of observations in the data and the subgroup $\mathcal{G}_j^k$.*

The proof of Theorem 2 is shown in “Appendix Appendix C”. Theorem 2 has the same practical implications as Theorem 1: Even if the features are independent, we will, in expectation, get the marginal PDPs. And when trees are grown deeper than needed, in expectation the cs-PDPs will yield the same curve.

Both the PDP and the set of cs-PDPs need $O(nM)$ evaluations, since $\sum_{k=1}^{K_j} n_k = n$ (and worst case $O(n^2)$ if evaluated at each $x_j^{(i)}$ value). Again, there is an additional cost for training the respective decision trees and making predictions.

5 Experiments

Since for real data sets there are no ground truth values for cPFI and cPDP available, we targeted a diverse set of metrics in our experiments:

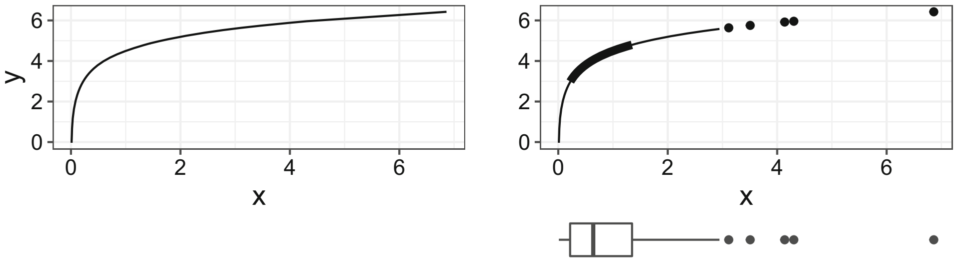


Fig. 4 Left: Marginal PDP. Bottom right: Boxplot showing the distribution of feature X. Top right: PDP with boxplot-like emphasis. In the x-range, the PDP is drawn from $\pm 1.58 \cdot IQR/\sqrt{n}$, where IQR is the range between the 25% and 75% quantile. If this range exceeds $[min(x_j), max(x_j)]$, the PDP is capped. Outliers are drawn as points. The PDP is bold between the 25% and 75% quantiles

- Conditional PFI Ground Truth Simulation: With this simulated experiment, we compared various cPFI methods. Since the data were simulated, we could compute the ground truth cPFI and benchmark all methods accordingly.
- Data fidelity evaluation: This experiment used real data sets to analyze how well the different perturbation methods that underpin the various cPDP/cPFI approaches avoid extrapolation.
- Model fidelity: This experiment evaluates how close the cPDP curves are to the real model predictions.

5.1 Training conditional sampling approaches

To ensure that sampling approaches are not overfitting, we suggest to separate training and sampling, where training covers all estimation steps that involve data. For this purpose, we refer to the training data with $\mathcal{D}_{train}$ and to the data for importance computation with $\mathcal{D}_{test}$. This section both describes how we compared the sampling approaches in the following chapters and serves as a general recommendation for how to use the sampling approaches.

For our cs-permutation, we trained the CART / transformation trees on $\mathcal{D}_{train}$ and permuted X_j of $\mathcal{D}_{test}$ within the terminal nodes of the tree. For CVIRF (Strobl et al. 2008; Debeer and Strobl 2020), which is specific to random forests, we trained the random forest on $\mathcal{D}_{train}$ to predict the target y and permuted X_j of $\mathcal{D}_{test}$ within the terminal nodes. For Model-X knockoffs (Candes et al. 2018), we fitted the second-order knockoffs on $\mathcal{D}_{train}$ and replaced X_j in $\mathcal{D}_{test}$ with its knockoffs. For the imputation approach (Fisher et al. 2019), we trained a random forest on $\mathcal{D}_{train}$ to predict X_j from X_{-j}, and replaced values of X_j in $\mathcal{D}_{test}$ with their random forest predictions plus a random residual. For the interval-based sampling (Apley and Zhu 2016), we computed quantiles of X_j using $\mathcal{D}_{train}$ and perturbed X_j in $\mathcal{D}_{test}$ by moving each observation once to the left and once to the right border of the respective intervals. The marginal permutation (PFI, PDP) required no training, we permuted (i.e., shuffled) the feature X_j in $\mathcal{D}_{test}$.

5.2 Conditional PFI ground truth simulation

We compared our cs-PFI approach using CART (tree cart) and transformation trees (tree trtr), CVIRF (Strobl et al. 2008; Debeer and Strobl 2020), Model-X knockoffs (ko) (Candes et al. 2018) and the imputation approach (impute rf) (Fisher et al. 2019) in ground truth simulations. We simulated the following data generating process: $y^{(i)} = f(\mathbf{x}^{(i)}) = \mathbf{x}_1^{(i)} \cdot \mathbf{x}_2^{(i)} + \sum_{j=1}^{10} x_j^{(i)} + \epsilon^{(i)}$, where $\epsilon^{(i)} \sim N(0, \sigma_\epsilon)$. All features, except feature X_1 followed a Gaussian distribution: $X_j \sim N(0, 1)$. Feature X_1 was simulated as a function of the other features plus noise: $x_1^{(i)} = h(x_{-1}^{(i)}) + \epsilon_x$. We simulated the following scenarios by changing h and ϵ_x:

- In the *independent* scenario, X_1 did not depend on any feature: $h(\mathbf{x}_{-1}^{(i)}) = 0$, $\epsilon_x \sim N(0, 1)$. This scenario served as a test how the different conditional PFI approaches handle the edge case of independence.
- The *linear* scenario introduces a strong correlation of X_1 with feature X_2: $h(\mathbf{x}_{-1}^{(i)}) = \mathbf{x}_2^{(i)}, \epsilon_x \sim N(0, 1)$.
- In the *non-linear* scenario, we simulated X_1 as a non-linear function of multiple features: $h(\mathbf{x}_{-1}^{(i)}) = 3 \cdot \mathbb{1}(\mathbf{x}_2^{(i)} > 0) - 3 \cdot \mathbb{1}(\mathbf{x}_2^{(i)} \leq 0) \cdot \mathbb{1}(\mathbf{x}_3^{(i)} > 0)$. Here also the variance of $\epsilon_x \sim N(0, \sigma_x)$ is a function of x: $\sigma_x(\mathbf{x}^{(i)}) = \mathbb{1}(\mathbf{x}_2^{(i)} > 0) + 2 \cdot \mathbb{1}(\mathbf{x}^{(i)} \leq 0) \cdot \mathbb{1}(\mathbf{x}_3^{(i)} > 0) + 5 \cdot \mathbb{1}(\mathbf{x}_2^{(i)} \leq 0) \cdot \mathbb{1}(\mathbf{x}_3^{(i)} \leq 0)$.
- For the *multiple linear dependencies* scenario, we chose X_1 to depend on many features: $h(\mathbf{x}_{-1}^{(i)}) = \sum_{j=2}^{10} x_j^{(i)}, \epsilon_x \sim N(0, 5)$.

For each scenario, we varied the number of sampled data points $n \in \{300, 3000\}$ and the number of features $p \in \{9, 90\}$. To "train" each of the cPFI methods, we used $2/3 \cdot n$ (200 or 2000) data points and the rest (100/1000) to compute the cPFI. The experiment was repeated 1000 times. We examined two settings.

- In setting (I), we assumed that the model recovered the true model $\hat{f} = f$.
- In setting (II), we trained a random forest with 100 trees (Breiman 2001).

In both settings, the true conditional distribution of X_1 given the remaining features is known (function h and error distribution is known). Therefore we can compute the ground truth conditional PFI, as defined in Eq. (2), by replacing $\hat{f}$ with f. We generated the samples of X_1 according to g to get the $\tilde{X}_1$ values and compute the increase in loss. The conditonal PFIs differed in settings (I) and (II) since in (I) we used the true f, and in (II) the trained random forest $\hat{f}$.

5.2.1 Conditional PFI ground truth results

For setting (I), the mean squared errors between the estimated conditional PFIs and the ground truth are displayed in Table 2, and the distributions of conditional PFI estimates in Fig. 5. In the *independent scenario*, where conditional and marginal PFI are equal, all methods performed equally well, except in the low n, high p scenario, where the knockoffs sometimes failed. As expected, the variance was higher for all methods when $n = 300$. In the *linear scenario*, the marginal PFI was clearly different from the conditional PFI. There was no clear best performing conditional PFI approach, as the results differ depending on training size n and number of features p. For low n and low p, knockoffs performed best. For high p, regardless of n, the cs-permutation approaches worked best, which might be due to the feature selection mechanism inherent to trees. The *multiple linear dependencies scenario* was the only scenario in which the cs-PFI approach was consistently outperformed by the other methods. Decision trees already need multiple splits for recovering linear relationships, and in this scenario, multiple linear relationships had to be recovered. Imputation with random forest worked well when multiple linear dependencies are present. For knockoffs, the results were mixed. As expected, the cs-PFI approach worked well in the *non-linear scenario*, and outperformed all other approaches. Knockoffs and imputation with random forests both overestimated the conditional PFI (except for knockoffs for $n = 300$ and $p = 90$). In addition to this bias, they had a larger variance compared to the cs-PFI approaches.

Generally, the transformation trees performed equal to or outperformed CART across all scenarios, except for the multiple linear dependencies scenario. Our cs-PFI approaches worked well in all scenarios, except when multiple (linear) dependencies were present. Even for a single linear dependence, the cs-PFI approaches were on par with knockoffs and imputation, and clearly outperformed both when the relationship was more complex.

In setting (II), a random forest was analyzed, which allowed us to include the conditional variable importance for random forests (CVIRF) by Strobl et al. (2008) and Debeer and Strobl (2020) in the benchmark. The MSEs are displayed in "Appendix Appendix D", Table 6, and the distribution of conditional PFI estimates in "Appendix Appendix D" in Fig. 11. The results for all other approaches are comparable to setting (I). For the low n settings, CVIRF worked as well as the other approaches in the *independent scenario*. It outperformed the other approaches in the *linear scenario* and the *multiple linear scenario* (when n was small). The CVIRF approach consistently underestimated the conditional PFI in all scenarios with high n, even in the *independent scenario*. Therefore, we would recommend to analyze the conditional PFI for random forests using cs-PFI for lower dimensional dependence structures, and imputation for multiple (linear) dependencies.

5.3 Trading interpretability for accuracy

In an additional experiment, we examined the trade-off between the depth of the trees and the accuracy with which we recover the true conditional PFI. For scenario (I), we trained decision trees with different maximal depths (from 1 to 10) and analyzed how the resulting number of subgroups influenced the conditional PFI estimate. The experiment was repeated 1000 times. The deeper the trees, the better the true conditional PFI was approximated. Also no overfitting occured, which is in line with theoretical considerations in Theorem 1. See "Appendix Appendix E" for detailed results.

Table 2 MSE comparing estimated and true conditional PFI (scenario I)

Setting	cs-PFI (cart)	cs-PFI (trtr)	impute rf	ko	mPFI
Independent					
n = 300, p = 10	1.33	1.35	1.67	1.47	1.39
n = 300, p = 90	1.50	1.29	1.46	5.81	1.31
n = 3000, p = 10	0.14	0.15	0.16	0.13	0.15
n = 3000, p = 90	0.15	0.14	0.14	0.18	0.13
Linear					
n = 300, p = 10	4.62	4.30	3.64	2.03	44.83
n = 300, p = 90	5.55	5.26	17.53	11.63	45.36
n = 3000, p = 10	0.40	0.26	0.26	0.63	37.40
n = 3000, p = 90	0.45	0.31	3.55	0.38	36.32
Multi. lin.					
n = 300, p = 10	2443.67	2623.54	1276.41	1583.69	2739.83
n = 300, p = 90	2574.54	2896.47	2141.01	6607.73	2988.68
n = 3000, p = 10	1031.83	900.68	140.98	810.78	1548.37
n = 3000, p = 90	1075.95	1041.10	438.25	185.13	1599.59
Non-linear					
n = 300, p = 10	22.00	17.76	265.73	668.34	1204.17
n = 300, p = 90	19.99	19.81	504.53	131.77	1248.74
n = 3000, p = 10	1.18	1.00	144.77	626.80	1156.32
n = 3000, p = 90	1.17	1.13	206.01	579.02	1136.83

Impute rf: Imputation with a random forest, ko: Model-X knockoffs, mPFI: (marginal) PFI, tree cart: cs-permutation based on CART, tree trtr: cs-permutation based on transformation trees

5.4 Data fidelity evaluation

PDP and PFI work by data intervention, prediction, and subsequent aggregation (Scholbeck et al. 2019). Based on data $\mathcal{D}$, the intervention creates a new data set. In order to compare different conditional sampling approaches, we define a measure of data fidelity to quantify the ability to preserve the joint distribution under intervention. Failing to preserve the joint distribution leads to extrapolation when features are dependent. Model-X knockoffs, for example, are directly motivated by preserving the joint distribution, while others, such as accumulated local effect plots do so more implicitly.

Data fidelity is the degree to which a sample $\tilde{X}_j$ of feature X_j preserves the joint distribution, that is, the degree to which $(\tilde{X}_j, X_{-j}) \sim (X_j, X_{-j})$ In theory, any measure that compares two multivariate distributions can be used to compute the data fidelity. In practice, however, the joint distribution is unknown, which makes measures such as the Kullback-Leibler divergence impractical. We are dealing with two samples, one data set without and one with intervention.

In this classic two-sample test-scenario, the maximum mean discrepancy (MMD) can be used to compare whether two samples come from the same distribution (Fortet

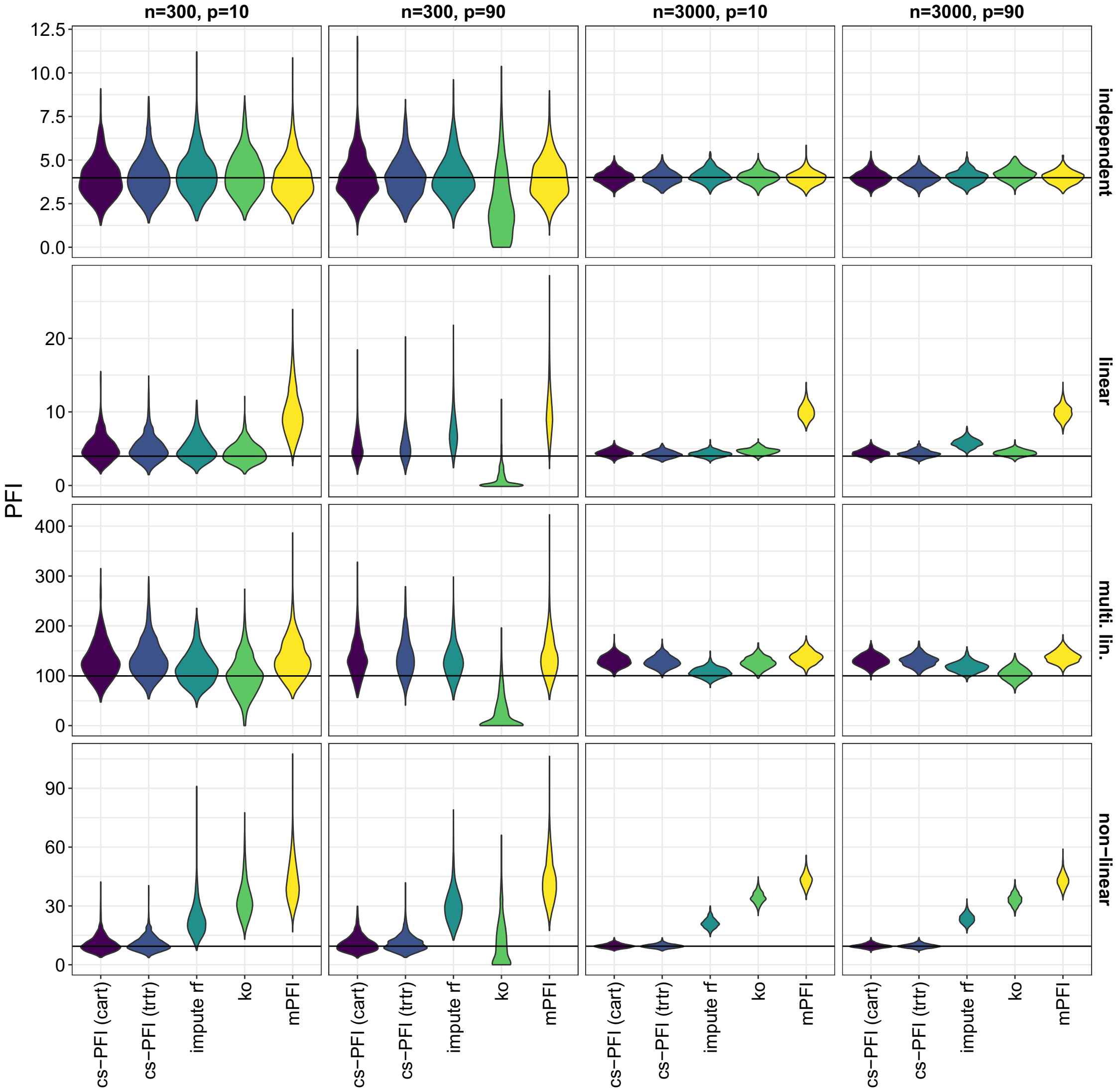


Fig. 5 Setting (I) comparing various conditional PFI approaches on the true model against the true conditional PFI (horizontal line) based on the data generating process

and Mourier 1953; Gretton et al. 2007, 2012; Smola et al. 2007). The empirical MMD is defined as:

$$\mathrm{MMD}(\mathcal{D}, \tilde{\mathcal{D}}) = \frac{1}{n^2} \sum_{x,z \in \mathcal{D}} k(x,z) - \frac{2}{nl} \sum_{x \in \mathcal{D}, z \in \tilde{\mathcal{D}}} k(x,z) + \frac{1}{l^2} \sum_{x,z \in \tilde{D}} k(x,z) \tag{6}$$

where $\mathcal{D} = \{x_j^{(i)}, x_{-j}^{(i)}\}_{i=1}^n$ is the original data set and $\tilde{\mathcal{D}} = \{\tilde{x}_j^{(i)}, x_{-j}^{(i)}\}_{i=1}^l$ a data set with perturbed $x_j^{(i)}$. For both data sets, we scaled numerical features to a mean of zero and a standard deviation of one. For the kernel k we used the radial basis function kernel for all experiments. For parameter σ of the radial basis function kernel, we chose the median L2-distance between data points which is a common heuristic (Gretton et al. 2012). We measure data fidelity as the negative logarithm of the MMD ($-log(\mathrm{MMD})$) to obtain a more condensed scale where larger values are better.

Definition 1 (*MMD-based Data Fidelity*) Let $\mathcal{D}$ be a dataset, and $\tilde{D}$ be another dataset from the same distribution, but with an additional intervention. We define the data fidelity as: Data Fidelity $= -log(\text{MMD}(\mathcal{D}, \tilde{\mathcal{D}}))$.

We evaluated how different sampling strategies (see Table 1) affect the data fidelity measure for numerous data sets of the OpenML-CC18 benchmarking suite (Bischl et al. 2019). We removed all data sets with 7 or fewer features and data sets with more than 500 features. See "Appendix Appendix F" for an overview of the remaining data sets. For each data set, we removed all categorical features from the analysis, as the underlying sampling strategies of ALE plots and Model-X knockoffs are not well equipped to handle them. We were foremost interested in two questions:

(A) How does cs-permutation compare with other sampling strategies w.r.t. data fidelity?
(B) How do choices of tree algorithm (CART vs. transformation trees) and tree depth parameter affect data fidelity?

In each experiment, we selected a data set, randomly sampled a feature and computed the data fidelity of various sampling strategies as described in the pseudo-code in Algorithm 2.

Algorithm 2: Data Fidelity Experiments

```
Input: OpenML-CC18 data sets, sampling strategies
for data set $\mathcal{D}$ in OpenML-CC18 do
    Remove prediction target from $\mathcal{D}$ (only keep it for CVIRF)
    Randomly order features in $\mathcal{D}$
    for features $j \in \{1, \dots, 10\}$ do
        for repetition $\in \{1, \dots, 30\}$ do
            Sample $min(10.000, n)$ rows from $\mathcal{D}$
            Split sample into $\mathcal{D}_{train}$ (40%), $\mathcal{D}_{test}$ (30%) and $\mathcal{D}_{ref}$ (30%)
            for each sampling do
                "Train" sampling approach using $\mathcal{D}_{train}$ (e.g., construct subgroups, fit
                knockoff-generator, ...)
                Generate conditional sample $\tilde{X}_j$ for $\mathcal{D}_{test}$
                Estimate data fidelity as $-log(MMD(\mathcal{D}_{ref}, \mathcal{D}_{test}))$
return Set of data fidelity estimates
```

For an unbiased evaluation, we split the data into three pieces: $\mathcal{D}_{train}$ (40% of rows), $\mathcal{D}_{test}$ (30% of rows) and $\mathcal{D}_{ref}$ (30% of rows). We used $\mathcal{D}_{train}$ to "train" each sampling method (e.g., train decision trees for cs-permutation, see Sect. 5.1). We used $\mathcal{D}_{ref}$, which we left unchanged and $\mathcal{D}_{test}$, for which the chosen feature was perturbed to estimate the data fidelity. For each data set, we chose 10 features at random, for which sampling was applied. Marginal permutation (which ignores the joint distribution) and "no perturbation" served as lower and upper bounds for data fidelity. For CVIRF, we only used one tree per random forest as we only compared the general perturbation strategy which is the same for each tree.

We repeated all experiments 30 times with different random seeds and therefore different data splits. All in all this produced 12,210 results (42 data sets × (up to) 10 features × 30 repetitions) per sampling method. All results are shown in detail in “Appendix Appendix F” (Figs. 13, 14, 15, and 16).

Since the experiments are repeated across the same data sets and the same features, the data fidelity results are not independent. Therefore, we used a random intercept model (Bryk and Raudenbush 1992) to analyze the differences in data fidelity between different sampling approaches. The target variable of the random intercept model was the MMD, the dependent variable was the perturbation method, and we used a random intercept per data and per feature (nested). So, informally: $MMD \sim$ perturbation method $+ (1|$ dataset/feature).

We chose “Marginal Permutation” as the reference category. We fitted two random intercept models: One to compare cs-permutation with fully-grown trees (CART, trtr) with other sampling methods and another one to compare different tree depths.

5.4.1 Results (A) state-of-the-art comparison

Figure 6 shows the effect estimates of different sampling approaches modeled with a random intercept model. The results show that cs-permutation performed better than all other methods. Model-X knockoffs and the imputation approach (with random forests) came in second place and outperformed ALE and CVIRF. Knockoffs were proposed to preserve the joint distribution, but are based on multivariate Gaussian distribution. This seems to be too restrictive for the data sets in our experiments. CVIRF does not have much higher data fidelity than marginal permutation. However, results for CVIRF must be viewed with caution, since data fidelity regards all features equally – regardless of their impact on the model prediction. For example, a feature can be highly correlated with the feature of interest, but might not be used in the random forest. A more informative experiment for comparing CVIRF can be found in Sect. 5.2. Figures 13 and 14 in “Appendix Appendix F” show the individual data fidelity results for the OpenML-CC18 data sets. Not perturbing the feature at all has the highest data fidelity and serves as the upper bound. The marginal permutation serves as a lower baseline. For most data sets, cs-permutation has a higher data fidelity compared to all other sampling approaches. For all the other methods there is at least one data set on which they reach a low data fidelity (e.g., “semeion”, “qsar-biodeg” for ALE; “nodel-simulation”, “churn” for imputation; “jm1”, “pc1” for knockoffs). In contrast, cs-permutation achieves a consistently high data fidelity on all these data sets.

Additionally, we review the data fidelity rankings of the sampling methods in Table 3. The table shows the average ranking of each method according to MMD. First we computed the rank of each perturbation method per dataset, feature and repetition, with rank 1 being the best (lowest MMD). This allows another view on the performance of the perturbation methods. The rankings show a similar picture as the random intercept model estimates, except that Model-X knockoffs have a better average ranking than imputation. This could be the case since on a few data sets (bank-marketing, electricity, see Fig. 13 in “Appendix Appendix F”) Model-X knock-

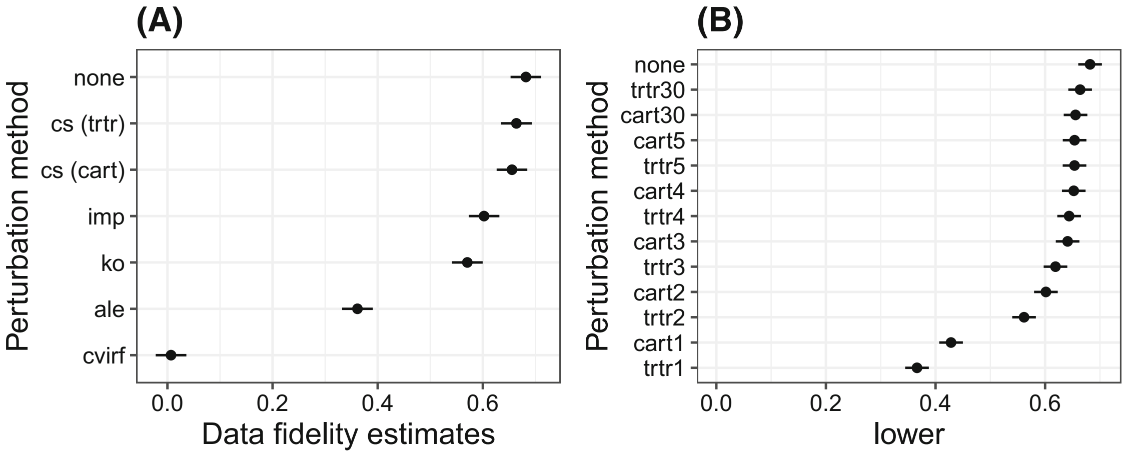


Fig. 6 Linear regression model coefficients and 95% confidence intervals for the effect of different sampling approaches on data fidelity, with (nested) random effects per data set and feature. **A** Comparing different sampling approaches. No perturbation ("none") and permutation ("perm") serve as upper and lower bounds. **B** Comparing cs-permutation using either CART or transformation trees and different tree depths (1, 2, 3, 4, 5 and 30). Marginal permutation is the reference category and therefore is at $x = 0$ and all other perturbation method estimates are relative to this reference

Table 3 Mean ranks and their standard deviation based on data fidelity of various perturbation methods over data sets, features and repetitions

	None	cs (trtr)	ko	cs (cart)	imp	ale	perm	cvirf
Mean ranks	2.50	3.51	3.70	3.76	4.25	4.61	6.82	6.84
SD	0.73	0.87	1.32	0.91	1.37	2.07	1.14	1.14

None: No intervention, which serves as upper benchmark. cart30: cs-permutation with CART with maximal depth of 30. trtr30: cs-permutation with transformation trees with maximal depth of 30. imp: Imputation approach. ko: Model-X knockoffs (Candes et al. 2018). ale: ALE perturbation (Apley and Zhu 2016). cvirf: Conditional variable importance for random forests (Strobl et al. 2008). perm: Unconditional permutation

offs have a very low data fidelity but on most others a higher model fidelity than the imputation method.

5.4.2 Results (B) tree configuration

We included shallow trees with maximum depth parameter from 1 to 5 to analyze the trade-off between tree depth and data fidelity. We included trees with a maximum depth parameter of 30 ("fully-grown" trees as this was the software's limit) as an upper bound for each decision tree algorithm. Figure 6B) shows that the deeper the trees (and the more subgroups), the higher the data fidelity. This is to be expected, since deeper trees allow for a more fine-grained separation of distributions. More importantly, we are interested in the trade-off between depth and data fidelity. Even splitting with a maximum depth of only 1 (two subgroups) strongly improves data fidelity over the simple marginal permutation for most data sets. A maximum depth of two means another huge average improvement in data fidelity, and already puts cs-permutation on par with knockoffs. A depth of three to four is almost as good as a maximum depth parameter of 30 and already outperforms all other methods, while still

Table 4 We selected data sets from OpenML Vanschoren et al. (2014) and Casalicchio et al. (2017) having 1000–8000 instances and a maximum of 50 numerical features

	wine	satellite	wind	space	pollen	quake
No. of rows	6497	6435	6574	3107	3848	2178
No. of features	12	37	15	7	6	4

We excluded data sets with categorical features, since ALE cannot handle them

being interpretable due to their shortness. CART slightly outperforms transformation trees clearly when trees are shallow, which is surprising since transformation trees are, in theory, better equipped to handle changes in the distribution. Deeply grown transformation trees (max. depth of 30) slightly outperform CART. Figures 15 and 16 in "Appendix Appendix F" show data fidelity aggregated by data set.

5.5 Model fidelity

Model fidelity has been defined as how well the predictions of an explanation method approximate the ML model (Ribeiro et al. 2016). Similar to Szepannek (2019), we define model fidelity for feature effects as the mean squared error between model prediction and the prediction of the partial function f_j (which depends only on feature X_j) defined by the feature effect method, for example $f_j(x) = PDP_j(x)$. For a given data instance with observed feature value $x_j^{(i)}$, the predicted outcome of, for example, a PDP can be obtained by the value on the y-axis of the PDP at the observed x_j value.

$$\text{Model\_Fidelity}(\hat{f}, f_j) = \frac{1}{n}\sum_{i=1}^{n}\left(\hat{f}\left(x^{(i)}\right) - f_j\left(x_j^{(i)}\right)\right)^2, \qquad (7)$$

where f_j is a feature effect function such as ALE or PDP. For this definition of model fidelity, lower values are more desirable. The better the model fidelity, the closer the effect curve is to the actual model predictions. In order to evaluate ALE plots, they have to be adjusted such that they are on a comparable scale to a PDP (Apley and Zhu 2016): $f_j^{ALE,adj} = f_j^{ALE} + \frac{1}{n}\sum_{i=1}^{n}\hat{f}(x^{(i)})$.

We trained random forests (500 trees), linear models and k-nearest neighbours models (k = 7) on various regression data sets (Table 4). 70% of the data were used to train the ML models and the transformation trees/CARTs. This ensure that results are not over-confident due to overfitting, see also Sect. 5.1. The remaining 30% of the data were used to evaluate model fidelity. For each model and each data set, we measured model fidelity between effect prediction and model prediction [Eq. (7)], averaged across observations and features.

Table 5 shows that the model fidelity of ALE and PDP is similar, while the cs-PDPs have the best model fidelity (lower is better). This is an interesting result since the decision trees for the cs-PDPs are neither based on the model nor on the real target, but solely on the conditional dependence structure of the features. However, the cs-PDPs have the advantage that we obtain multiple plots. We did not aggregate the plots to

Table 5 Mean model fidelity averaged over features in a random forest for various data sets, and the variance across features

	Pollen	Quake	Satellite	Space	Wind	Wine
PDP	10.83 (6.33)	0.03 (0.0)	4.78 (0.03)	0.04 (0.0)	43.98 (33.91)	0.75 (0.0)
ALE	12.33 (19.68)	0.04 (0.0)	4.82 (0.01)	0.04 (0.0)	43.38 (56.71)	0.75 (0.0)
trtr1	9.09 (3.18)	0.03 (0.0)	4.19 (0.59)	0.04 (0.0)	30.36 (41.02)	0.72 (0.0)
cart1	9.06 (3.24)	0.03 (0.0)	3.75 (0.77)	0.04 (0.0)	31.22 (59.23)	0.72 (0.0)
trtr2	8.29 (5.14)	0.03 (0.0)	3.36 (0.52)	0.04 (0.0)	26.47 (50.21)	0.71 (0.0)
cart2	8.12 (6.29)	0.03 (0.0)	3.23 (0.69)	0.04 (0.0)	27.29 (78.63)	0.71 (0.0)

The cPDPs (trtr,cart) always had a lower loss (i.e. higher model fidelity) than PDP and ALE. The loss monotonically decreases with increasing maximum tree depth for subgroup construction

a single conditional PDP, but computed the model fidelity for the PDPs within the subgroups (visualized in Fig. 9). Our cs-PDPs using trees with a maximum depth of 2 have a better model fidelity than using a maximum depth of 1. We limited the analysis to interpretable conditioning and therefore allowed only trees with a maximum depth of 2, since a tree depth of 3 already means up to 8 subgroups which is already an impractical number of PDPs to have in one plot. CART sometimes beats trtr (e.g., on the "satellite" data set) but sometimes trtr has a lower loss (e.g., on the "wind" data set). Using different models (knn or linear model) produced similar results, see "Appendix Appendix G".

6 Application

In the following application, we demonstrate that cs-PDPs and cs-PFI are valuable tools to understand model and data beyond insights given by PFI, PDPs, or ALE plots. We trained a random forest to predict daily bike rentals (Dua and Graff 2017) with given weather and seasonal information. The data ($n = 731$, $p = 9$) was divided into 70% training and 30% test data. The features are not independent (see "Appendix Appendix H")

6.1 cs-PDPs and cs-PFI

To construct the subgroups, we used transformation trees with a maximum tree depth of 2 which limited the number of possible subgroups to 4. We chose transformation trees because they are theoretically more sound and don't require the assumption that the conditional distributions only differ in the means of the other features.

Figure 7 shows that for most features the biggest change in the estimated conditional PFI happens when moving from a maximum depth of 0 (= marginal PFI) to a depth of 2. This makes a maximum depth of 2 a reasonable trade-off between limiting the number of subgroups and accurately approximating the conditional PFI. We compared the marginal and conditional PFI for the bike rental predictions, see Fig. 8.

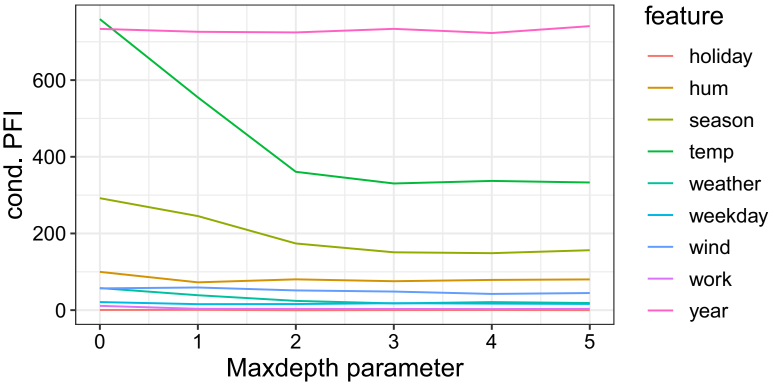


Fig. 7 Conditional feature importance by increasing maximum depth of the trees

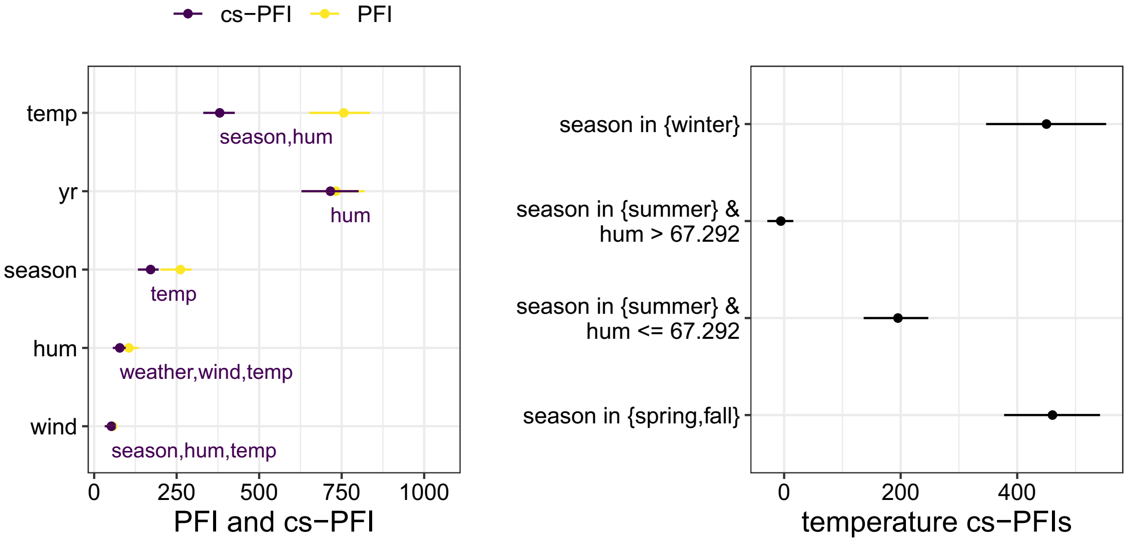


Fig. 8 Left: Comparison of PFI and cs-PFI for a selection of features. For cs-PFI we also show the features that constitute the subgroups. Right: Local cs-PFI of temperature within subgroups. The temperature feature is important in spring, fall and winter, but neglectable on summer days, especially humid ones

The most important features, according to (marginal) PFI, were temperature and year. For the year feature, the marginal and conditional PFI are the same. Temperature is less important when we condition on season and humidity. The season already holds a lot of information about the temperature, so this is not a surprise. When we know that a day is in summer, it is not as important to know the temperature to make a good prediction. On humid summer days, the PFI of temperature is zero. However, in all other cases, it is important to know the temperature to predict how many bikes will be rented on a given day. The disaggregated cs-PFI in a subgroup can be interpreted as "How important is the temperature, given we know the season and the humidity".

We compare PDP, ALE and cs-PDP in Fig. 9. Both ALE and PDP show a monotone increase of predicted bike rentals up until a temperature of 25 °C and a decrease beyond that. The PDP shows a weaker negative effect of very high temperatures which might be caused by extrapolation: High temperature days are combined with e.g. winter. A limitation of the ALE plot is that we should only interpret it locally within each interval that was used to construct the ALE plot. In contrast, our cs-PDP is explicit about the subgroup conditions in which the interpretation of the cs-PDP is valid and shows the

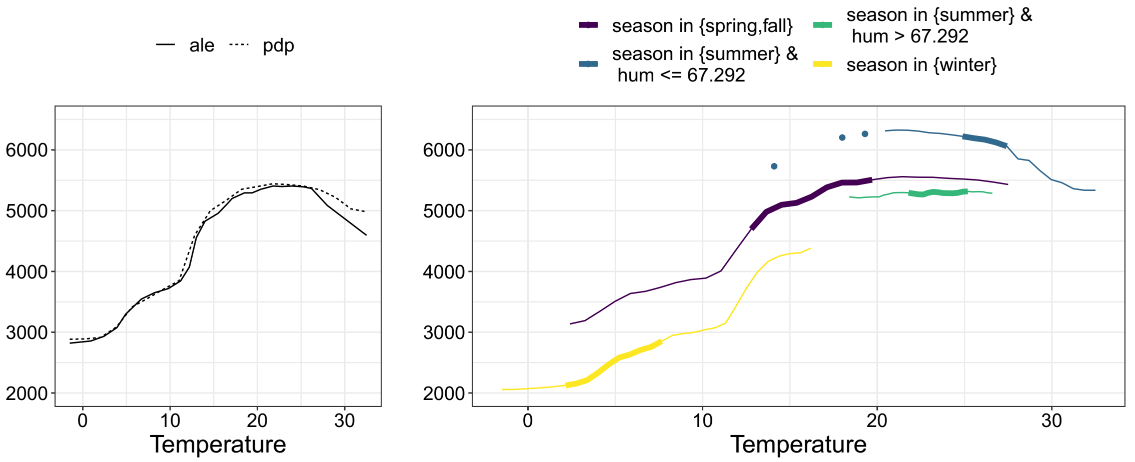


Fig. 9 Effect of temperature on predicted bike rentals. Left: PDP and ALE plot. Right: cs-PDPs for 4 subgroups

distributions in which the feature effect may be interpreted. The local cs-PDPs in subgroups reveal a more nuanced picture: For humid summer days, the temperature has no effect on the bike rentals, and the average number of rentals are below that of days with similar temperatures in spring, fall and drier summer days. The temperature has a slightly negative effect on the predicted number of bike rentals for dry summer days (humidity below 67.3). The change in intercepts of the local cs-PDP can be interpreted as the effect of the grouping feature (season). The slope can be interpreted as the temperature effect within a subgroup.

We also demonstrate the local cs-PDPs for the season, a categorical feature. Figure 10 shows both the PDP and our local cs-PDPs. The normal PDP shows that on average there is no difference between spring, summer and fall and only slightly less bike rentals in winter. The PDP with four subgroups conditional on temperature shows that the marginal PDP is misleading. The PDP indicates that in spring, summer and fall, around 4500 bikes are rented and in winter around 1000 fewer. The cs-PDPs in contrast show that, conditional on temperature, the differences between the seasons are much greater, especially for low temperatures. Only at high temperatures is the number of rented bikes similar between seasons.

7 Discussion

We proposed the cs-PFIs and cs-PDPs, wich are variants of PFI and PDP that work when features are dependent. Both cs-PFIs and cs-PDPs rely on permutations in subgroups based on decision trees. The approach is simple: Train a decision tree to predict the feature of interest and compute the (marginal) PFI/PDP in each terminal node defined by the decision tree.

Compared to other approaches, cs-PFIs and cs-PDPs enable a human comprehensible grouping, which carries information how dependencies affect feature effects and importance. As we showed in various experiments, our methods are on par or outperform other methods in many dependence settings. We therefore recommend

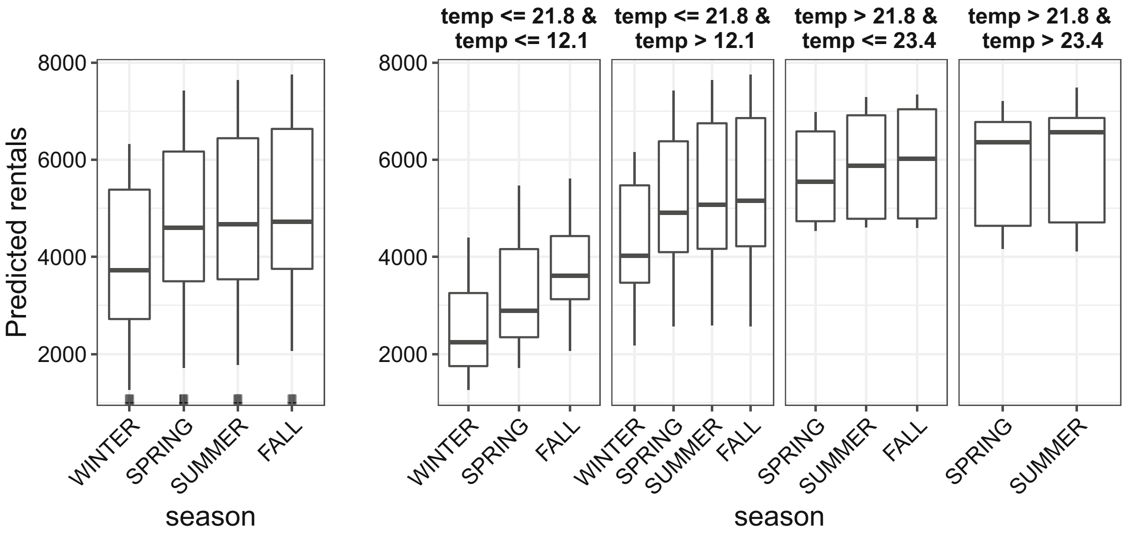


Fig. 10 Effect of season on predicted rentals. Left: PDP. Right: Local cs-PDPs. The cs-PDPs are conditioned on temperature, in which the tree split at 21.5 and at 9.5

using cs-PDPs and cs-PFIs to analyze feature effects and importances when features are dependent. However, due to their construction with decision trees, cs-PFIs and cs-PDPs do not perform well when the feature of interest depends on many other features, but only if it depends on a few features. Especially the interpretability suffers if the tree has to rely on many features. We recommend analyzing the dependence structure beforehand, using the imputation approach with random forests in the case of multiple dependencies, and cs-PFIs in all other cases.

Our framework is flexible regarding the choice of partitioning and we leave the evaluation of the rich selection of possible decision tree and decision rules approaches to future research.

Reproducibility All experiments were conducted using *mlr* (Lang et al. 2019) and R (R Core Team 2017). We used the *iml* package (Molnar et al. 2018) for ALE and PDP, *party*/*partykit* (Hothorn and Zeileis 2015) for CVIRF and *knockoff* (Patterson and Sesia 2020) for Model-X knockoffs. The code for all experiments is available at https://github.com/christophM/paper_conditional_subgroups.

Acknowledgements This project is funded by the Bavarian State Ministry of Science and the Arts, by the Bavarian Research Institute for Digital Transformation (bidt), and supported by the German Federal Ministry of Education and Research (BMBF) (Grant No. 01IS18036A) and by the German Research Foundation (DFG), Emmy Noether Grant 437611051. The authors of this work take full responsibilities for its content.

Funding Open Access funding enabled and organized by Projekt DEAL.

Chapter 3

Discussion

3.1 Our Contributions and the Three Challenges

At the beginning of this thesis, we postulated three challenges: Unclear explananda (Challenge I), misinterpretation (Challenge II), and the estimation of conditional-sampling-based techniques (Challenge III).

As follows, I summarise how our contributions helped in tackling the aforementioned challenges. Furthermore, I discuss the limitations of our contributions and open problems.

3.1.1 Challenge I: Unclear Explananda

We argued that interpretations are consulted in various contexts and for various goals. As a consequence, the requirements that we have towards interpretation methods are conflicting, meaning that no interpretation method can address them all.

Thus, to make progress, we have to disentangle the different goals and contexts such that they inform coherent requirements. Then, given a fixed goal with coherent requirements, the *explanandum* can be determined and formalised.

Over the course of this thesis, we inspected two interpretation contexts in more detail: recourse and inference.

- *Recourse:* In Paper I, we disentangled two goals that were previously conflated. Wachter et al. [2017a] propose to use counterfactual explanations for three purposes: To understand decisions, to contest

decisions, and to change decisions via recourse. We argue that these goals conflict: To understand and contest decisions, we have to take the model's prediction $\hat{Y}$ into account. In contrast, recourse should focus on improvement and thus concerns the underlying target Y. Only by separating recourse from related goals such as contestability, we were liberated to focus recourse recommendations on the underlying target Y.
The distinction between nine perspectives on model and data helped us to articulate the differences between the conflicting subgoals contestability and recourse and to express the new target estimand.

- *Scientific Inference:* Papers II and III are concerned with scientific inference. They are motivated by the observation that many methods originally developed to describe the model's mechanism are used to gain insight into the DGP. These goals are in conflict, and thus we separate scientific inference from other interpretation purposes.
  Scientific inference itself is motivated by various contexts and goals. Separating those in detail would require significant domain knowledge and is thus beyond the scope of this thesis. However, we propose a general procedure that scientific practitioners can follow to derive the explanandum for a specific interpretation goal in the context of scientific inference. Moreover, we provide several examples of explananda that may be of interest when interpreting ML models to learn about the associations in the DGP.

Open Problems and Outlook

Our contributions helped separate two interpretation goals from other conflicting goals: recourse and inference. Each of these interpretation goals can be refined further, for instance, by assessing how the targeted improvement confidence should be chosen, by disentangling subtasks in scientific inference or by focusing on causal inference with IML.
Furthermore, there are common motivations for IML that our work did not tackle and that require clarification:

- *Robustness:* A particularly prominent motivation for IML is to assess robustness. But robustness to what? In principle, various shifts are conceivable [Bühlmann, 2020, Mohus and Li, 2023, Lee et al., 2021, Perdomo et al., 2020, Hancox-Li, 2020]. Even for the same shift, different explananda are conceivable: For example, we could explain the model's mechanism so accurately that the explainee can predict the model's behaviour outside the distribution, or we could directly simulate the shift and quantify how that impacts the model's performance.

- *Contestability:* Furthermore, IML is often motivated to help assess whether a model's decision-making aligns with moral, ethical and legal standards. When the standards are violated, the explainee shall be enabled to contest the model's reasoning and thereby revert the decision. However, moral, ethical and legal standards are diverse, and thus they must be treated separately. For example, a range of different fairness definitions exist [Barocas et al., 2019].

3.1.2 Challenge II: Misinterpretation

In the general introduction we argued that practitioners are often confused about the meaning of IML methods and base their choice of method on superficial criteria such as publication year. We argued that the choice of method should be determined by the interpretation goal such that an explanatory link between explanandum and explanation must be established. To enable practitioners to establish such a link, we need interpretation rules that clarify what insight methods can and cannot provide.
Furthermore, the estimation of the techniques involves uncertainties which may result in misleading estimates. Thus, the involved uncertainties should be quantified and communicated to the explainee.

Contributions

- In §3, we introduce a taxonomy of nine different perspectives on model and data. The taxonomy has proven to help describe explananda and understand the meaning of explanations. Thus, it can serve as one connecting link between the interpretation goal and the interpretation method.
- In Paper I, we provide interpretation rules for contrastive explanations, clarifying under which circumstances *acceptance* and *improvement* align. More specifically, we find that under interventions on non-causal variables, targeting acceptance may not lead to improvement but to

actions that game the predictor. This aligns with our distinction between the effects of data-level causation on the prediction and the underlying target (§3). However, given that there are no unobserved confounders, interventions on causes of the prediction target leave the conditional distribution of the target given the covariates intact, such that *acceptance guarantees* can be derived from improvement guarantees.

Furthermore, instead of only giving a recommendation, we suggest communicating the recommendations along with the probabilities of them leading to improvement or acceptance. As such, we make a first step towards quantifying and communicating the uncertainties involved in recourse.

- In Paper II, we clarify that many IML methods are unsuitable for learning about the data since they are concerned with understanding individual model elements or the model's mechanism within and outside the distribution. In general, individual model elements and the model's behaviour outside the training distribution cannot be linked to properties DGP. In contrast, conditional-sampling-based methods can be used to gain insight into the DGP since they leverage that models are holistically representational, meaning that the model as a whole represents an aspect of the distribution.

- In Paper III, we show how various estimation uncertainties affect the interpretation, especially when the interpretation is used to learn about the DGP. Moreover, we propose variance estimands and confidence intervals that practitioners can use to inform their conclusions.

- In Paper IV, we generalise PFI and CFI to a class of feature importance algorithms called Relative Feature Importance (RFI). We derive interpretation rules for each member, clarifying what conclusions about the dependence structure in the data and the model's mechanism can be drawn from nonzero feature importance.

- In Paper V, we raise awareness for a broad range of interpretation pitfalls, including assuming one-fits-all interpretability, interpreting models that do not generalise well, ignoring feature dependence and feature interactions, ignoring model and approximation uncertainty, and unjustified causal interpretation.

Open Problems and Outlook

There are a range of open challenges that we leave for future work:

- *Robustness of Acceptance Guarantees:* In Paper I, we quantify the uncertainties stemming from our inability to perfectly predict Y or the effects of interventions. However, further uncertainties influence whether someone will improve or get accepted.
  For example, we do not assess the long-term effects of recourse on the decision model and how they affect the acceptance guarantees. It is conceivable that the conditional distribution of the target given the covariates may be altered by including recourse-implementing individuals. In preliminary work, we demonstrate that the model's mechanism may change significantly when refitted on mixed pre- and post-recourse datasets [König et al., 2021].
  If such uncertainties are not considered, acceptance guarantees may be overly optimistic.

- *Uncertainty stemming from conditional sampling:* In Paper II, we argued that for inference about the associations in the data, conditional-sampling-based methods should be used. However, they require access to a conditional sampler, which usually is not readily available, such that the sampler must be learned.
  In Paper III, we quantify various uncertainties involved in estimating IML methods. However, we neglect uncertainty stemming from learning the conditional sampler. Quantifying this uncertainty is difficult since each fit may correlate not only with refits of the sampler but also with refits of the model. Thus, simply refitting the sampler and measuring the variance yields biased estimates.

- *Interpreting the Magnitude of RFI:* In Paper IV, we provide interpretation rules for what implications can be drawn from nonzero feature importance. The conclusions tell us *whether* we can conclude that dependence is present. However, they do not allow us how to interpret the magnitude of the importance scores. Interpretation rules that take the magnitude into account are an interesting direction for future work.

3.1.3 Challenge III: Estimation of Conditional-Sampling-Based Techniques

Although easy to compute, interpretation techniques that are based on marginal perturbations only allow limited insight into the DGP. For example, we learned that nonzero PFI does not necessarily imply that the feature is in any way related to the underlying target. Instead, the nonzero importance could stem from dependencies between the feature and its covariates (Paper IV). As an alternative, conditional-sampling-based techniques such as CFI were proposed.

The problem with conditional-sampling-based techniques is that they are more difficult to estimate. Conditional samplers are in general, not readily available, and modelling conditional distributions is a challenging problem. Throughout the thesis, we propose two techniques that help make the estimation of conditional-sampling-based techniques computationally more efficient.

Contributions

- In Paper VI, we leverage causal structure learning to greedily identify conditional independencies in the data, allowing us to make SAGE estimation more efficient by skipping expensive to compute surplus value function evaluations. We achieve significant runtime gains since the one-time effort of learning a graph that encodes the independence structure is negligible compared to the cost of the many saved value function evaluations.

- In Paper VII, we propose to leverage tree-based learners to learn a partitioning of the feature space such that the features within each partition are (close to) independent. Thus, within partitions marginal and conditional sampling coincide. We leverage the partitioning to sample from conditional distributions and to estimate CFI. Furthermore, in settings where the partitioning is sparse and thus interpretable, subgroup-specific interpretations can be generated that provide novel insight into model and data.

Open Problems and Outlook

Although the papers help in making the estimation of conditional-sampling-based techniques more efficient, they suffer from limitations:

- For the subgroup-specific versions of PFI and PDP in Paper VII, we assume a sparse discrete structure in the data. Although the method performs on par with competing approaches on continuous data, the partitioning trees become deeper, such that the subgroup-specific outputs become challenging to interpret.
- In Paper VI, we exploit knowledge of the dependence structure in the data to skip unnecessary evaluations. The final SAGE value estimates are nearly unaffected. It would be interesting to assess whether knowledge of the dependence structure in the data can be used to improve the quality of the approximation, for instance, by reducing conditioning sets and partitioning multivariate distributions, thereby reducing the dimensionalities of the estimation problems.
- For the computation of SAGE values, we have to estimate multivariate conditional distributions, potentially with mixed continuous and categorical variables. To the best of my knowledge, the only general-purpose solution is to decompose the multivariate conditional into a sequence of conditionals [Bates et al., 2021]. I conjecture that the approximation quality deteriorates quickly in high-dimensional settings since error accumulates.
- For many conditional-sampling-based methods, alternatives exist that refit the model but do not need conditional samplers. More work is required to study under which circumstances refitting-based approaches should be preferred over conditional-sampling-based approaches.
  For example, both Leave-One-Covariate-Out Importance (LOCO) [Lei et al., 2018], a refitting-based technique, and CFI provide similar insight into the dependence structure of the data. One may argue that LOCO does not explain the model and thus provides qualitatively different insight than CFI; However, in many scenarios such as inference, we do not care about explaining a specific model in the first place. In these settings, using CFI only makes sense if refitting the model is more difficult or expensive than learning a conditional sampler.
  We leave a detailed comparison for future work.

3.2 Further Challenges

Causal Explanations With Limited Causal Knowledge

In Paper I, we demonstrate how to generate Improvement-Focused Recourse Recommendations (ICR) in settings where the causal graph or the Structural Causal Model (SCM) is known. However, in practice, neither may be available or the causal assumptions may be violated.

As a next step, it would be interesting to assess how partial knowledge of the causal graph can be used to guide recourse. For instance, if we know that variable x_1 is a direct cause of y, we may be able to estimate the respective treatment effects irrespective of whether some other variable x_2 is a cause or a confounded variable. More generally, given partial causal knowledge, effect bounds may be derived [Maathuis et al., 2010].

Furthermore, when individuals implement recourse recommendations, the resulting changes provide a valuable signal about the causal structure in the data [Bechavod et al., 2020]. Using this signal, wrong causal assumptions may be corrected in the long term.

Furthermore, recourse recommendations could be designed to provide insight into the causal structure in the data, e.g., using methods from the active learning literature [Sussex et al., 2021]. However, such an intervention must be carefully assessed from an ethical perspective.

Feedback Loops

In IML it is often assumed that the explanation does not affect the DGP. However, as feedback loops may invalidate the explanation. Let us illustrate this at the example of recourse.

Suppose a decision-making system is used to distribute a limited good. For instance, there are limits to how much money a bank can lend, or to how many jobs a company can offer.

In such scenarios, the decision boundary must be calibrated to account for the limited availability of the resource. For instance, the qualification level required to land a job depends on who else applies.

Current research in recourse typically assumes binary decision problems and a fixed 0.5 decision boundary. If many people implement recourse, the decision boundary may be shifted, and a resource recommendation that works for a 0.5 decision threshold may be outdated once the recourse action is implemented.

This can be seen as a further advantage of focusing on improvement; However, it poses a significant challenge for communicating recourse recommendations. Should we simply freeze the decision boundary for individuals at the point in time when they requested the recommendation to make accurate guarantees? Under what circumstances do we reach an equilibrium state where the decision boundary no longer shifts? Can we predict the change of decision boundary over time?
We leave an investigation of those questions for future work.

Learning Human-Intelligible Concepts

In this thesis, we focused on model-agnostic interpretation techniques and tabular data. In tabular data settings, the variables (usually) correspond to meaningful concepts that the explainee can reason about. However, in many relevant ML applications, they don't. For example, in computer vision, the model inputs are pixels; but humans don't reason on the pixel level.
A range of work aims to explain models and data in terms of concepts. For example, Koh et al. [2020] adapt the model architecture to predict human-generated concept labels such that the model's prediction can be explained in terms of these concepts. [Bau et al., 2017] try to understand the meaning of individual neurons by correlating their activations with concept labels. Kim et al. [2018] use exemplary images with a concept to quantify the model's sensitivity to the concept. Goyal et al. [2019] estimate the causal effect of labelled concepts on the prediction.
Problematically, the aforementioned methods require concept labels or concept examples. However, such supervision may not be readily available in practice. Furthermore, except for Koh et al. [2020], we do not know whether the provided concepts reflect the model's reasoning.
A promising avenue of research leverages work on causal abstraction [Rubenstein et al., 2017, Beckers and Halpern, 2019, Beckers et al., 2020, Markham and Grosse-Wentrup, 2020] to learn abstract representations of the model's mechanism [Geiger et al., 2021, 2023a,b, Wu et al., 2021, 2023].
The problem with existing methods is that they focus on abstracting the model, but many tasks in IML are not concerned with the model's mechanism per se (§1.4.3). Given that dependencies in the data are neglected, it is questionable whether the learned concepts are useful for goals such as inference or recourse. We leave a detailed investigation for future work.

Human Studies

In their seminal paper, Doshi-Velez and Kim [2017] discuss that IML methods can be evaluated on three levels: Application-grounded evaluation assigns real humans real tasks that they are supposed to solve using the explanations. Human-grounded metrics also involve real humans but assess the performance in simplified tasks. For functionally grounded evaluation, a formal definition of “interpretability” is used to compute the quality of the explanation.

Eventually, what we care about is whether explainees are enabled to draw the correct conclusions by inspecting the explanations. In my view, experiments with real humans are thus the gold standard for assessing the quality of explanations.

However, human experiments are costly. Furthermore, human studies are not required to narrow down the choice of method: Given a fixed interpretation goal the explanandum can be determined and the explanations that do not concern the explanandum can be ruled out (functional-grounded evaluation). For instance, when using IML for inference, we can formally assess whether an explanation actually describes the desired property of the DGP.

In my view, a functional grounded evaluation should precede an application-grounded evaluation. To design an application-grounded evaluation, we need to clarify the goal anyway. Furthermore, it is conceivable that to assess whether the study participants draw the correct conclusions, the explanandum must be defined as well.

Thus our work so far is only concerned with functionally grounded evaluation. This is not to say that an application-grounded evaluation may provide interesting new insights. For example, in recourse, it would be interesting to assess how we should communicate recommendations and guarantees given the involved uncertainties. We leave a detailed investigation for future work.

Chapter 4

Conclusion: If Interpretability Is the Answer, What is the Question?

As we argued throughout this thesis, it is unclear what interpretability means. Interpretability is associated with a range of conflicting goals, and interpretation techniques provide a range of conflicting answers. When provided with an explanation, it can thus be difficult to pin down what question was asked or what goal was pursued in the first place.

When interpreting ML systems, we should start with what we eventually care about: the interpretation goal. The interpretation goal determines the question and must thus inform the choice of IML methods. Throughout this thesis, we demonstrated the importance of following this order. Only by fixing the interpretation goal first, we could argue about how to design recourse explanations and which methods to choose for scientific inference.

Unfortunately, in practice, IML methods are often not chosen based on their suitability to answer a specific question but instead based on superficial criteria such as publication year [Krishna et al., 2022]. I speculate that – despite efforts to expose misconceptions in the field [Lipton, 2018, Páez, 2019, Freiesleben and König, 2023] – the assumption that "interpretability" is a monolithic goal persists. As a result of this faulty reasoning, the generated explanations may not match the question and thus degrade to mere justifications. I hope that with our work, we contribute a step towards a responsible application of IML.

Given that interpretability is more of an umbrella term for several goals than a goal in itself, I am optimistic that in the long run, "interpretability" will be gradually replaced with more precise terminology and that the IML community will regroup along the goals and questions that motivated the field in the first place.

Bibliography

Daniel W Apley and Jingyu Zhu. Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 82(4):1059–1086, 2020.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning: Limitations and Opportunities*. fairmlbook.org, 2019. http://www.fairmlbook.org.

Stephen Bates, Emmanuel Candès, Lucas Janson, and Wenshuo Wang. Metropolized knockoff sampling. *Journal of the American Statistical Association*, 116(535):1413–1427, 2021.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549, 2017.

Yahav Bechavod, Katrina Ligett, Zhiwei Steven Wu, and Juba Ziani. Causal feature discovery through strategic modification. *arXiv preprint arXiv:2002.07024*, 3, 2020.

Sander Beckers and Joseph Y Halpern. Abstracting causal models. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pages 2678–2685, 2019.

Sander Beckers, Frederick Eberhardt, and Joseph Y Halpern. Approximate causal abstractions. In *Uncertainty in artificial intelligence*, pages 606–615. PMLR, 2020.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623, 2021.

Kristin Blesch, David S Watson, and Marvin N Wright. Conditional feature importance for mixed data. *AStA Advances in Statistical Analysis*, pages 1–20, 2023.

Stephan Bongers, Patrick Forré, Jonas Peters, and Joris M Mooij. Foundations of structural causal models with cycles and latent variables. *The Annals of Statistics*, 49(5):2885–2915, 2021.

Sebastian Bordt, Michèle Finck, Eric Raidl, and Ulrike von Luxburg. Post-hoc explanations fail to achieve their purpose in adversarial contexts. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 891–905, 2022.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Peter Bühlmann. Invariance, causality and robustness. 2020.

Hugh Chen, Joseph D Janizek, Scott Lundberg, and Su-In Lee. True to the model or true to the data? *arXiv preprint arXiv:2006.16234*, 2020.

Ian Covert, Scott M Lundberg, and Su-In Lee. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33:17212–17223, 2020.

Marina Danilevsky, Kun Qian, Ranit Aharonov, Yannis Katsis, Ban Kawas, and Prithviraj Sen. A survey of the state of explainable ai for natural language processing. *arXiv preprint arXiv:2010.00711*, 2020.

Devleena Das, Siddhartha Banerjee, and Sonia Chernova. Explainable ai for robot failures: Generating explanations that improve user assistance in fault recovery. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, pages 351–360, 2021.

Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE symposium on security and privacy (SP)*, pages 598–617. IEEE, 2016.

Ashley Deeks. The judicial demand for explainable artificial intelligence. *Columbia Law Review*, 119(7):1829–1850, 2019.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

Heather E Douglas. Reintroducing prediction to explanation. *Philosophy of Science*, 76(4):444–463, 2009.

Hadi Fanaee-T. Bike Sharing Dataset. UCI Machine Learning Repository, 2013. DOI: https://doi.org/10.24432/C5W894.

Jean-Marc Fellous, Guillermo Sapiro, Andrew Rossi, Helen Mayberg, and Michele Ferrante. Explainable artificial intelligence for neuroscience: behavioral neurostimulation. *Frontiers in neuroscience*, 13:1346, 2019.

Krzysztof Fiok, Farzad V Farahani, Waldemar Karwowski, and Tareq Ahram. Explainable artificial intelligence for education and training. *The Journal of Defense Modeling and Simulation*, 19(2):133–144, 2022.

Timo Freiesleben. The intriguing relation between counterfactual explanations and adversarial examples. *Minds and Machines*, pages 1–33, 2021.

Timo Freiesleben and Gunnar König. Dear xai community, we need to talk! fundamental misconceptions in current xai research. *arXiv preprint arXiv:2306.04292*, 2023.

Timo Freiesleben, Gunnar König, Christoph Molnar, and Alvaro Tejero-Cantero. Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena. *arXiv preprint arXiv:2206.05487*, 2022.

Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

Christopher Frye, Damien de Mijolla, Tom Begley, Laurence Cowton, Megan Stanley, and Ilya Feige. Shapley explainability on the data manifold. *arXiv preprint arXiv:2006.01272*, 2020.

Krishna Gade, Sahin Cem Geyik, Krishnaram Kenthapadi, Varun Mithal, and Ankur Taly. Explainable ai in industry. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3203–3204, 2019.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. *Advances in Neural Information Processing Systems*, 34:9574–9586, 2021.

Atticus Geiger, Chris Potts, and Thomas Icard. Causal abstraction for faithful model interpretation. *arXiv preprint arXiv:2301.04709*, 2023a.

Atticus Geiger, Zhengxuan Wu, Christopher Potts, Thomas Icard, and Noah D Goodman. Finding alignments between interpretable causal variables and distributed neural representations. *arXiv preprint arXiv:2303.02536*, 2023b.

Caroline M Gevaert. Explainable ai for earth observation: A review including societal and regulatory perspectives. *International Journal of Applied Earth Observation and Geoinformation*, 112:102869, 2022.

Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *journal of Computational and Graphical Statistics*, 24 (1):44–65, 2015.

Lauren Gordon, Teodor Grantcharov, and Frank Rudzicz. Explainable artificial intelligence for safe intraoperative decision support. *JAMA surgery*, 154(11): 1064–1065, 2019.

Yash Goyal, Amir Feder, Uri Shalit, and Been Kim. Explaining classifiers with causal concept effect (cace). *arXiv preprint arXiv:1907.07165*, 2019.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42, 2018.

Leif Hancox-Li. Robustness in machine learning explanations: Does it matter? In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 640–647, 2020.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Paul W Holland. Statistics and causal inference. *Journal of the American statistical Association*, 81(396):945–960, 1986.

Andreas Holzinger, Anna Saranti, Christoph Molnar, Przemyslaw Biecek, and Wojciech Samek. Explainable ai methods-a brief overview. In *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*, pages 13–38. Springer, 2020.

Giles Hooker and Lucas Mentch. Please stop permuting features: An explanation and alternatives. *arXiv preprint arXiv:1905.03151*, 2, 2019.

Dominik Janzing, Lenon Minorics, and Patrick Blöbaum. Feature relevance quantification in explainable ai: A causal problem. In *International Conference on artificial intelligence and statistics*, pages 2907–2916. PMLR, 2020.

José Jiménez-Luna, Francesca Grisoni, and Gisbert Schneider. Drug discovery with explainable artificial intelligence. *Nature Machine Intelligence*, 2(10): 573–584, 2020.

Amir-Hossein Karimi, Julius Von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse under imperfect causal knowledge: a probabilistic approach. *Advances in Neural Information Processing Systems*, 33: 265–277, 2020a.

Amir-Hossein Karimi, Julius von Kügelgen, Bernhard Schölkopf, and Isabel Valera. Towards causal algorithmic recourse. In *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers*, pages 139–166. Springer, 2020b.

Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse: from counterfactual explanations to interventions. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 353–362, 2021.

Hassan Khosravi, Simon Buckingham Shum, Guanliang Chen, Cristina Conati, Yi-Shan Tsai, Judy Kay, Simon Knight, Roberto Martinez-Maldonado, Shazia Sadiq, and Dragan Gašević. Explainable artificial intelligence in education. *Computers and Education: Artificial Intelligence*, 3:100074, 2022.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques.* MIT press, 2009.

Gunnar König, Timo Freiesleben, and Moritz Grosse-Wentrup. A causal perspective on meaningful and robust algorithmic recourse. *arXiv preprint arXiv:2107.07853*, 2021.

Gunnar König, Timo Freiesleben, and Moritz Grosse-Wentrup. Improvement-focused causal recourse (icr). In *AAAI*, 2023.

Konstantina Kourou, Themis P Exarchos, Konstantinos P Exarchos, Michalis V Karamouzis, and Dimitrios I Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and structural biotechnology journal*, 13:8–17, 2015.

Satyapriya Krishna, Tessa Han, Alex Gu, Javin Pombra, Shahin Jabbari, Steven Wu, and Himabindu Lakkaraju. The disagreement problem in explainable machine learning: A practitioner's perspective. *arXiv preprint arXiv:2202.01602*, 2022.

Maya Krishnan. Against interpretability: a critical examination of the interpretability problem in machine learning. *Philosophy & Technology*, 33(3): 487–502, 2020.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1–8, 2019.

Jae-Gil Lee, Yuji Roh, Hwanjun Song, and Steven Euijong Whang. Machine learning robustness, fairness, and their convergence. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 4046–4047, 2021.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113(523):1094–1111, 2018.

Konstantinos G Liakos, Patrizia Busato, Dimitrios Moshou, Simon Pearson, and Dionysis Bochtis. Machine learning in agriculture: A review. *Sensors*, 18(8):2674, 2018.

Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. *Applied Stochastic Models in Business and Industry*, 17(4):319–330, 2001.

Peter Lipton. Contrastive explanation. *Royal Institute of Philosophy Supplements*, 27:247–266, 1990.

Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3): 31–57, 2018.

Helen E Longino. *The fate of knowledge.* Princeton University Press, 2018.

Robert WP Luk. A theory of scientific study. *Foundations of Science*, 22(1): 11–38, 2017.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

Christoph Luther, Gunnar König, and Moritz Grosse-Wentrup. Efficient sage estimation via causal structure learning. In *AISTATS*, 2023.

Marloes H Maathuis, Diego Colombo, Markus Kalisch, and Peter Bühlmann. Predicting causal effects in large-scale systems from observational data. *Nature methods*, 7(4):247–248, 2010.

R Machlev, L Heistrene, M Perl, KY Levy, J Belikov, S Mannor, and Y Levron. Explainable artificial intelligence (xai) techniques for energy and power systems: Review, challenges and opportunities. *Energy and AI*, 9:100169, 2022.

Alex Markham and Moritz Grosse-Wentrup. Measurement dependence inducing latent causal models. In *Conference on Uncertainty in Artificial Intelligence*, pages 590–599. PMLR, 2020.

A Miguel, ROBINS HERNAN, and M JAMES. *Causal Inference: What If.* CRC PRESS, 2023.

Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

Tim Miller. Contrastive explanation: A structural-model approach. *The Knowledge Engineering Review*, 36:e14, 2021.

Mathias Lundteigen Mohus and Jinyue Li. Adversarial robustness in unsupervised machine learning: A systematic review. *arXiv preprint arXiv:2306.00687*, 2023.

Christoph Molnar. *Interpretable machine learning.* Lulu. com, 2020.

Christoph Molnar, Timo Freiesleben, Gunnar König, Giuseppe Casalicchio, Marvin N Wright, and Bernd Bischl. Relating the partial dependence plot and permutation feature importance to the data generating process. *arXiv preprint arXiv:2109.01433*, 2021.

Christoph Molnar, Gunnar König, Julia Herbinger, Timo Freiesleben, Susanne Dandl, Christian A Scholbeck, Giuseppe Casalicchio, Moritz Grosse-Wentrup, and Bernd Bischl. General pitfalls of model-agnostic interpretation methods for machine learning models. In *xxAI-Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*, pages 39–68. Springer, 2022.

Christoph Molnar, Gunnar König, Bernd Bischl, and Giuseppe Casalicchio. Model-agnostic feature importance and effects with dependent features: a conditional subgroup approach. *Data Mining and Knowledge Discovery*, pages 1–39, 2023.

Andrés Páez. The pragmatic turn in explainable artificial intelligence (xai). *Minds and Machines*, 29(3):441–459, 2019.

Judea Pearl. *Causality*. Cambridge university press, 2009.

Judea Pearl and Dana Mackenzie. *The book of why: the new science of cause and effect*. Basic books, 2018.

Juan Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *International Conference on Machine Learning*, pages 7599–7609. PMLR, 2020.

Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

Manish Raghavan, Solon Barocas, Jon Kleinberg, and Karen Levy. Mitigating bias in algorithmic hiring: Evaluating claims and practices. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 469–481, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450369367.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016. doi: 10.1145/2939672.2939778.

Paul K Rubenstein, Sebastian Weichwald, Stephan Bongers, Joris M Mooij, Dominik Janzing, Moritz Grosse-Wentrup, and Bernhard Schölkopf. Causal consistency of structural equation models. *UAI*, 2017.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

Wesley C Salmon. Why ask,'why?'? an inquiry concerning scientific explanation. In *Hans Reichenbach: logical empiricist*, pages 403–425. Springer, 1979.

Lloyd S Shapley et al. A value for n-person games. 1953.

Deepak Kumar Sharma, Jahanavi Mishra, Aeshit Singh, Raghav Govil, Gautam Srivastava, and Jerry Chun-Wei Lin. Explainable artificial intelligence for cybersecurity. *Computers and Electrical Engineering*, 103:108356, 2022.

Galit Shmueli et al. To explain or to predict? *Statistical science*, 25(3):289–310, 2010.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, prediction, and search.* MIT press, 2000.

Carolin Strobl, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. Conditional variable importance for random forests. *BMC bioinformatics*, 9(1):1–11, 2008.

Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41:647–665, 2014.

Scott Sussex, Caroline Uhler, and Andreas Krause. Near-optimal multi-perturbation experimental design for causal structure learning. *Advances in Neural Information Processing Systems*, 34:777–788, 2021.

Chakkrit Kla Tantithamthavorn and Jirayus Jiarpakdee. Explainable ai for software engineering. In *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 1–2. IEEE, 2021.

Adi L Tarca, Vincent J Carey, Xue-wen Chen, Roberto Romero, and Sorin Drăghici. Machine learning and its applications to biology. *PLoS computational biology*, 3(6):e116, 2007.

Akif B Tosun, Filippo Pullara, Michael J Becich, D Taylor, Jeffrey L Fine, and S Chakra Chennubhotla. Explainable ai (xai) for anatomic pathology. *Advances in Anatomic Pathology*, 27(4):241–250, 2020.

Berk Ustun, Alexander Spangher, and Yang Liu. Actionable recourse in linear classification. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 10–19, 2019.

Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. Why a right to explanation of automated decision-making does not exist in the general data protection regulation. *International Data Privacy Law*, 7(2):76–99, 2017a.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017b.

David S Watson. Conceptual challenges for interpretable machine learning. *Synthese*, 200(1):1–33, 2022.

David S Watson and Marvin N Wright. Testing conditional independence in supervised learning algorithms. *Machine Learning*, 110(8):2107–2129, 2021.

Adrian Weller. Transparency: motivations and challenges. In *Explainable AI: interpreting, explaining and visualizing deep learning*, pages 23–40. Springer, 2019.

James Woodward and Lauren Ross. Scientific Explanation. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Metaphysics Research Lab, Stanford University, Summer 2021 edition, 2021.

Zhengxuan Wu, Atticus Geiger, Josh Rozner, Elisa Kreiss, Hanson Lu, Thomas Icard, Christopher Potts, and Noah D Goodman. Causal distillation for language models. *arXiv preprint arXiv:2112.02505*, 2021.

Zhengxuan Wu, Karel D'Oosterlinck, Atticus Geiger, Amir Zur, and Christopher Potts. Causal proxy models for concept-based model explanations. In *International Conference on Machine Learning*, pages 37313–37334. PMLR, 2023.

Thorsten Wuest, Daniel Weimer, Christopher Irgens, and Klaus-Dieter Thoben. Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, 4(1):23–45, 2016.

Christopher C Yang. Explainable artificial intelligence for predictive modeling in healthcare. *Journal of healthcare informatics research*, 6(2):228–239, 2022.

Jiaming Zeng, Berk Ustun, and Cynthia Rudin. Interpretable classification models for recidivism prediction. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 180(3):689–722, 2017.

Qingyuan Zhao and Trevor Hastie. Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, 39(1):272–281, 2021.

Xingyu Zhou, Yuling Jiao, Jin Liu, and Jian Huang. A deep generative approach to conditional sampling. *Journal of the American Statistical Association*, pages 1–12, 2022.

List of Abbreviations

Abbreviations Used in the Thesis

CART	Classification and Regression Trees
CE	Counterfactual Explanation
CFI	Conditional Feature Importance
CR	Causal Recourse
DGP	Data-Generating Process
ER	Elementwise Representationality
HR	Holistic Representationality
ICE	Individual Conditional Expectation
ICR	Improvement-focused Causal Recourse
IML	Interpretable Machine Learning
MC	Monte Carlo
ML	Machine Learning
MSE	Mean Squared Error
PDP	Partial Dependence Plot
PFI	Permutation Feature Importance
RF	Random Forest
RL	Reinforcement Learning
SAGE	Shapley Additive Global importancE
SCM	Structural Causal Model
SHAP	SHapley Additive exPlanations
SVM	Support Vector Machine
XAI	eXplainable Artificial Intelligence

Supplements

Supplements for Paper I: Improvement-focused Causal Recourse (ICR)

A Extended Background

As follows, we recapitulate well-known definitions in our notation, provide more detailed background on related work and recapitulate results that we use in the proofs. Readers who are already familiar with recourse terminology and d-separation (A.1 and A.2), and who are not interested in more detailed introductions of intervention stability (A.3, only required for the proof of Proposition 2) or causal recourse (A.4), may skip this section.

A.1 Overview of important terms

An overview of important terms is provided in Table 1.

A.2 d-separation

Two variable sets X, Y are called d-separated [Geiger et al., 1990, Spirtes et al., 2000] by the variable set Z in a graph $\mathcal{G}$ (denoted as $X \perp_{\mathcal{G}} Y|Z$), if, and only if, for every path p it either holds that (i) p contains a chain $i \rightarrow m \rightarrow j$ or a fork $i \leftarrow m \rightarrow j$ where $m \in Z$ or (ii) p contains a collider $i \rightarrow m \leftarrow j$ such that m and for all of its descendants n it holds that $m, n \notin Z$. Given the causal Markov property, d-separation in a causal graph implies (conditional) independence in the data [Peters et al., 2017].

A.3 Generalizability and intervention stability

For Proposition 2, we leverage necessary conditions for invariant conditional distributions as derived in [Pfister et al., 2021]. The authors introduce a d-separation based intervention stability criterion that is applied to a modified version of $\mathcal{G}$. For every intervened upon variable X_l an auxiliary intervention variable, denoted as I_l, is added as direct cause of X_l, yielding $\mathcal{G}^*$. The intervention variable can be seen as a switch between different mechanisms. A set $S \subseteq \{1, \dots, d\}$ is called *intervention stable* regarding a set of actions if for all intervened upon variables X_l (where $l \in I^{\text{total}}$) the d-separation $I^l \perp_{\mathcal{G}^*} Y|X_S$ holds in $\mathcal{G}^*$. The authors show that intervention stability implies an invariant conditional distribution, i.e., for all actions $a, b \in \mathbb{A}$ with $I^a, I^b \subseteq I^{\text{total}}$ it holds that $p(y^a|x_S) = p(y^b|x_S)$ (Pfister et al. [2021], Appendix A).

A.4 Causal recourse

ICR is closely related to the CR framework [Karimi et al., 2020b, 2021], but differs substantially in its motivation and target. In order to allow for a direct comparison we briefly sketch the main ideas and the central CR definitions in our notation. Like ICR, CR aims to guide individuals to revert unfavorable algorithmic decisions (recourse). Therefore, they suggest to search for cost-efficient actions that lead to acceptance by the prediction model. Actions are modeled as structural interventions $a : \Pi \rightarrow \Pi$, which can be constructed as $a = do(\{X_i := \theta_i\}_{i \in I})$, where I is the index set of features to be intervened upon [Karimi et al., 2021]. The conservativeness of the suggested actions can be adjusted using the hyperparameter γ_{LCB}, that determines the adaptive threshold $\texttt{thresh}(a)$ and thereby how many standard deviations the expected prediction shall be away from the model's decision threshold t. In order to accommodate different levels of causal knowledge, two probabilistic versions of CR were introduced [Karimi et al., 2020b]: While individualized recourse assumes knowledge of the SCM, subpopulation-based CR only assumes knowledge of the causal graph.

Table 1: Overview of important terms and their meanings.

term	meaning
explainee	individual for whom the explanation is generated, e.g. loan applicant
model authority	decision-making entity, e.g. credit institute
recourse	action of the explainee that reverts unfavorable decision
acceptance	desirable model prediction ($\hat{Y} = 1$)
improvement	(yield) desirable state of the underlying target ($Y = 1$)
gaming	yield acceptance without improvement, e.g. treating the symptoms
pre-/post-recourse	before/after implementing recourse recommendation
contestability	the explainee's ability to contest an algorithmic decision
robustness of recourse	probability that recourse is accepted despite model/data shifts

Individualized recourse Individualized recourse predicts the effect of actions using structural counterfactuals [Karimi et al., 2021], which require a full specification of the SCM.

Given a function that evaluates the cost of actions (cost(a, x^{pre})), the optimization goal for individualized causal recourse is given below. The adaptive threshold `thresh` bounds the prediction away from the decision threshold.[18]

$$a^* \in \underset{a \in \mathbb{A}}{\text{argmin}} \quad \text{cost}(a, x^{pre}) \quad \text{s.t. } \mathbb{E}[\hat{h}(x^{post})|do(a), x^{pre}] \geq \texttt{thresh}(a)$$

$$\text{with } \texttt{thresh}(a) := 0.5 + \gamma_{LCB}\sqrt{\text{Var}[\hat{h}(x^{post,a})]}$$

Subpopulation-based recourse: If no knowledge of the SCM is given, counterfactual distributions cannot be estimated and consequently individualized recourse recommendations cannot be computed. Subpopulation-based CR is based on the average treatment effect within a subgroup of similar individuals [Karimi et al., 2020b]. More specifically individuals belong to the same group if the non-descendants $nd(I)$ of intervention variables (which ceteris paribus remain constant despite the intervention) take the same value. The subpopulation-based objective is given below.

$$a^* \in \underset{a \in \mathbb{A}}{\text{argmin}} \, \text{cost}(a, x^{pre}) \text{ s.t. } \mathbb{E}_{X_{d(I)}|do(X_I=\theta), x^{pre}_{nd(I)}}[\hat{h}(x^{pre}_{nd(I)}, \theta, X_{d(I)})] \geq \texttt{thresh}(a).$$

[18] Further constraints have been suggested, e.g., $x^{post,a} \in \mathcal{P}$lausible or $a \in \mathcal{F}$easible [Laugel et al., 2019, Ustun et al., 2019, Mahajan et al., 2020, Dandl et al., 2020, Karimi et al., 2021].

B Estimation and Optimization

As follows we provide detailed explanations of the proposed estimation procedures. First, we explain how to sample from the individualized post-recourse distribution, which allows us to estimate the individualized improvement and acceptance rates (γ^{ind} and η^{ind}, B.1). Based on the same sampling mechanism we can also estimate the individualized post-recourse prediction $h^{*,ind}$ (B.2). Then we explain how to sample from the subpopulation-based post-recourse distribution, which allows us to estimate the subpopulation-based improvement and acceptance rates (γ^{sub} and η^{sub}, B.3). Furthermore, we provide details on optimization (B.4) and demonstrate that the optimal observational predictor h^* can also be estimated using the SCM (B.5).

B.1 Estimation of the individualized improvement confidence γ^{ind} and individualized acceptance rate η^{ind}

We recall that γ^{ind} is the counterfactual probability of the underlying target Y taking the favorable outcome, and η^{ind} the counterfactual probability of the prediction $\hat{Y}$ taking the favorable outcome. In order to estimate γ^{ind} and η^{ind} we first sample covariates and target from the counterfactual post-recourse distribution and then compute the proportion of favorable outcomes for Y and $\hat{Y}$ in the sample.
In general, sampling from counterfactual distributions based on a SCM is performed in three steps (Section 3, [Pearl, 2009]).

1. *Abduction*: The exogenous noise variables are reconstructed from the observations, i.e., $p(u_{Y,D}|x^{pre})$ is estimated.
2. *Intervention*: The intervention $do(a)$ on the SCM $\mathcal{M}$ is performed by replacing the respective structural equations $f_{I_a} := \theta_{I_a}$, yielding $\mathcal{M}_{do(a)}$.
3. *Prediction*: The abducted noise variables are sampled from $p(u_{Y,D}|x^{pre})$ and passed through the model $\mathcal{M}_{do(a)}$ to sample from the counterfactual distribution $P(Y^{post}, X^{post}|x^{pre}, do(a))$.

Given knowledge of the SCM, the challenge is to sample the exogeneous variables from $p(u_{Y,D}|x^{pre})$ (abduction). As follows we explain the abduction in two steps. First, we explain how we can abduct u_j for variables for which both the node x_j and all parents $(x, y)_{pa(j)}$ are observed, which we refer to as the standard abduction case. Then we factorize the abduction of the joint $p(u_{Y,D}|x^{pre})$ into several components which can be reduced to said standard abduction case. The sampling procedure is summarized in Algorithm 1.

B.1.1 Recap: Standard abduction

If for a node u_j both the node $(x, y)_j$ and the parents $(x, y)_pa(j)$ are observed, we can apply standard abduction. The standard abduction procedure depends on the type of structural equation and exogenous noise distribution.
Given invertible structural equations, observation of $x_j, x_{pa(j)}$ determines u_j. More specifically, u_j can be reconstructed using

$$u_j = f^{-1}(x_j; x_{pa(j)}).$$

For instance, for additive structural equations $f_j(u_j; x_{pa(j)}) = g(x_{pa(j)}) + u_j$, the inversion is given by $f_j^{-1}(x_j; x_{pa(j)}) = x_j - g(x_{pa(j)})$.
In our experiments we also included binomial variables with a sigmoidal (non-invertible) structural equation. More specifically, the structural equations are defined as $x_j = [\sigma(l(x_{pa(j)})) \leq u_j]$ with $U_j \sim Unif(0, 1)$. Here σ refers to the sigmoid function and l to some linear combination. $[cond]$ evaluates to 1 when the condition is true and otherwise to 0. Intuitively, $\sigma(l(x_{pa(j)}))$ can be seen as a nonlinear activation function which determines the probability of the node being activated ($x_j = 1$). u_j acts as a dice, where values $\leq \sigma(l(x_{pa(j)}))$ imply $x_j = 1$ and vice versa.
For those variables, if $x_j = 1$, we know that $u_j \leq \sigma(l(x_{pa(j)}))$ and vice versa, such that we can abduct U_j as follows (and can therefore sample u_j):

$$P(U_j|x_j; x_{pa(j)}) = \begin{cases} Unif(0, \sigma(l(x_{pa(j)}))), & \text{for } x_j = 1 \\ Unif(\sigma(l(x_{pa(j)})), 1), & \text{for } x_j = 0 \end{cases}$$

As we will see in the next section, our estimation procedure can be flexibly extended to SCMs with different types of structural equations, as long as a procedure to sample from the abducted exogneous noise variable for the standard case (where parents and the node itself are observed) is available.

Algorithm 1: Sampling from the individualized post-recourse distribution

```
Data: pre-recourse observation $x^{pre}$, action $a$ (where $do(a) := do(X_{I_a} := \theta)$), sample size $M$, structural causal
      model $\mathcal{M}$ with structural equations $f_j$, observational predictor $h$
Result: sample from $p(y^{post}, x^{post}|x^{pre}, do(a))$
get $\mathcal{M}_{do(a)}$ by updating $f_i(x_{pa(i)}; u_i) := \theta_i$ for $i \in I_a$ ;
for $m$ in $(0, ..., M-1)$ do
    sample $y'$ from $Binomial(h(x^{pre}))$ ;
    for $j$ in $D$ do
        sample $u_j^{(m)}$ from $p(u_j|(x,y')_j, (x,y')_{pa(j)})$        ▷ comment: leveraging standard abduction;
    end
    sample $u_Y^{(m)}$ from $p(u_Y|y', x_{pa(Y)})$ ;
    compute $(x^{post}, y^{post})^{(m)} = f_{\mathcal{M}_{do(a)}}(u^{(m)})$ ;
end
```

B.1.2 Factorization of $p(u|x)$

We have demonstrated how to abduct individual nodes in the standard setting where the corresponding endogenous variable and its parents are observed.

As follows we demonstrate how to sample from the joint distribution of the exogenous variables given an observation of X (and without observing Y). Therefore, we show that $p(u|x)$ can be seen as a mixture of two distributions, one for each possible state y' of Y. In order to sample from it, we (1) need to sample y' from the mixing distribution $p(y|x)$ and (2) given y', sample from the respective abducted noise variable $p(u|y', x)$.

$$p(u|x) \overset{\text{law tot. prob.}}{=} \sum_{y'\in\{0,1\}} p(u, y'|x) \overset{\text{cond. prob.}}{=} \sum_{y'\in\{0,1\}} p(u|y', x)p(y'|x) \tag{2}$$

The binomial mixing distribution $p(y|x)$ can be obtained and sampled from by leveraging the cross-entropy optimal predictor h^* (which can for instance be derived from the SCM, see B.5). In order to sample from $p(u|y', x)$ we leverage the Markov factorization, which allows us to sample each component independently using the standard abduction procedure described above.

$$p(u|x, y') \overset{\text{d-sep.}}{=} P(u_Y|x_{pa(Y)}, y') \prod_{k\in ch(Y)} P(u_k|x_k, x_{pa(k)}, y') \prod_{k\notin ch(Y)} P(u_k|x_k, x_{pa(k)}). \tag{3}$$

The overall procedure is summarized in Algorithm 1.

B.1.3 Estimation of γ^{ind} and η^{ind}

Given the procedure to sample from the individualized post-recourse distribution we can estimate γ^{ind} by taking the mean over the samples taken for Y^{post}. Similarly, for each sample for X^{post} we can compute the prediction $\hat{y}^{post}$ using either $h \geq t$ or $h^{ind} \geq t$. By taking the mean over all sampled predictions $\hat{y}^{post}$ we can estimate the respective acceptance probability $\eta(t; x^{pre}, a, h)$ or $\eta(t; x^{pre}, a, h^{ind})$.

B.2 Estimation of the individualized post-recourse prediction

We continue to show how the individualized post-recourse prediction can be estimated. We recall that $h^{*,ind}$ is

$$h^{*,ind}(x^{post}; x^{pre}, a) = P(Y^{post} = 1|x^{post}, x^{pre}, do(a)).$$

We can estimate $h^{*,ind}$ by leveraging the procedure to sample from the post-recourse covariate distribution (Algorithm 1). More specifically, we draw samples (y', x') from $P(Y^{post}, X^{post}|do(a), x^{pre})$ and keep those that conform with x^{post} (i.e., $x' = x^{post}$). Within the subsample, we compute the proportion of samples for which $y' = 1$ to estimate $p(y^{post}|x^{pre}, x^{post}, do(a))$. In more formal terms, we approximate Eq. 4 using rejection sampling and Monte Carlo integration [Koller and Friedman, 2009].

If the structural equations are invertible[19] or the nodes are categorical the procedure is tractable, since many or all

[19] Meaning that the abducted joint distribution has point mass probability for two configurations, one for each possible state of Y.

Algorithm 2: Estimating $h^{*,ind}$

```
Data: pre-recourse observation $x^{pre}$, action $a$, sample size $M$, structural causal model $\mathcal{M}$, observational predictor
      $h, m = 0$
Result: $\hat{h}^{ind}(x^{post}; x^{pre}, do(a))$
while $m < M$ do
    sample $(x', y')$ using Alg. 1 and $x^{pre}, a, \mathcal{M}, h$;
    if $x' = x^{post}$ then
        $m = m + 1$; store $y'$ as $y'^{(m)}$ ;
    end
end
$\hat{h}^{ind}(x^{post}) = \frac{1}{M} \sum_{m=1}^{M} y'^{(m)}$
```

samples conform with x^{post}. Otherwise the estimation may become intractable. We see the application of likelihood weighting or MCMC as promising directions and refer interested readers to Koller and Friedman [2009].
In addition to the sampling-based procedure we also derive a closed-form solution for settings with invertible structural equations, which is provided in Proposition 5, Eq. 5.

Proposition 5. *In general, the individualized post-recourse predictor can be estimated as*

$$\begin{aligned} & p(y^{post}|x^{pre}, x^{post}, do(a)) \\ & = \frac{\int_{\mathcal{U}} p(y^{post}, x^{post}|u, do(a))p(u|x^{pre})du}{\sum_{y' \in \{0,1\}} \left(\int_{\mathcal{U}} p(y', x^{post}|u, do(a))p(u|x^{pre})du \right)} \end{aligned} \tag{4}$$

Given invertible structural equations, the individualized post-recourse prediction function reduces to

$$\begin{aligned} & p(y^{post}|x^{post}, x^{pre}, do(a)) \\ & = \frac{p(U_{-I} = f^{-1}_{do(a)}(y^{post}, x^{post})|x^{pre}, do(a))}{\sum_{y' \in \{0,1\}} p(U_{-I} = f^{-1}_{do(a)}(y', x^{post})|x^{pre}, do(a))}. \end{aligned} \tag{5}$$

B.3 Estimation of the subpopulation-based improvement confidence γ^{sub} and the subpopulation-based acceptance rate η^{sub}

As follows we detail how to estimate γ^{sub} and η^{sub}. We focus on actions a that potentially affect Y, meaning that they intervene on causes of Y.[20]
In order to estimate γ^{sub} and η^{sub} we sample (x', y') from the subpopulation-based post-recourse distribution. Given a sample from the subpopulation-based post-recourse distribution we can estimate γ^{sub} and η^{sub} by taking the respective sample means.
We explain the sampling procedure in two steps: We first recall how causal graphs can be leveraged to sample interventional distributions, and then explain why we can apply the procedure to sample from the subpopulation-based post-recourse distribution.

Recap: Sampling interventional distributions leveraging a causally sufficient causal graph $\mathcal{G}$ Given a causal graph $\mathcal{G}$ (that fulfills the global Markov property), the joint distribution $P(X, Y)$ can be reformulated using the Markov factorization, which makes use of the d-separations in the graph.

$$p(x, y) = p(y|x_{pa(y)}) \prod_{j \in D} p(x_j|(x, y)_{pa(j)})$$

As a consequence, we can sample from the joint distribution by sampling each component given its respective parents. In order to ensure that the parents for each node have been sampled already, the graph is traversed in topological order, starting with the root node and ending with the sink nodes [Koller and Friedman, 2009].
Given that causal sufficiency (no unobserved confounders) and the principle of independent mechanisms hold, the same

[20] Actions that do not affect Y trivially do not lead to improvement. The respective probability of $Y = 1$ can be estimated using the optimal observational predictor.

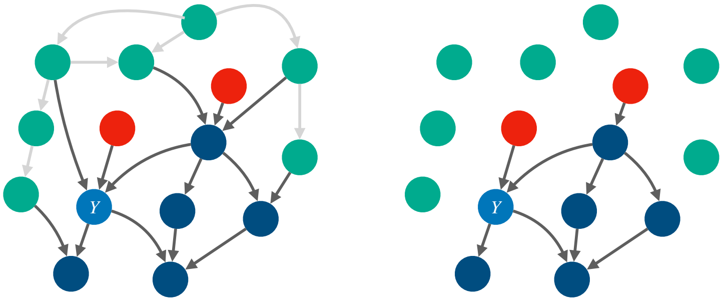


Figure 3: Causal graph $\mathcal{G}_{\overline{I_a}}$ visualizing the subpopulation-based post-recourse setting, including the prediction target Y (light blue), intervened-upon variables I_a (red), the subgroup characteristics G_a (cyan) and the descendants Γ that shall be resampled (dark blue). $\overline{I_a}$ indicates that incoming edges to I_a were removed. Right: Causal graph $\mathcal{G}_{\overline{I_a}\underline{G_a}}$ where incoming edges to I_a and outgoing edges from G_a were removed. We observe that in this manipulated graph G_a is d-separated from Γ. Thus, according to the second rule of do-calculus, for G_a intervention and conditioning coincide.

Algorithm 3: Sampling from the subpopulation-based post-recourse distribution

```
Data: pre-recourse observation $x^{pre}$, action $a$ with $I_a \cap asc(Y) \neq \emptyset$ ($do(a) := do(X_{I_a} := \theta)$), sample size $M$,
      causal graph $\mathcal{G}$, conditional distributions $P(X_j|X_{pa(j)})$ for $j \in \Gamma$ with $\Gamma := \{r : r \in asc(Y) \wedge r \in d(I)\}$
Result: sample from $p(y, x_\Gamma|do(a), x_{G_a})$
for $m \leftarrow 0$ to $M$ do
    $\Gamma^{sorted} \leftarrow$ topologicalsort( $\Gamma; \mathcal{G}_{do(a)}$)          ▷ sort such that causes precede effects ;
    for $j$ in $\Gamma^{sorted}$ do
        sample $(x, y)_j^{post,(m)} \sim P((X, Y)_j|(X, Y)_{pa(j)} = (x, y)_{pa(j)}^{post})$ ;
    end
end
```

procedure can also be applied when sampling from interventional distributions of the form $p(x, y|do(a))$ by leveraging the so-called truncated factorization. The intervened upon nodes are not sampled from their parents, but fixed to the values θ_a. The remaining nodes Γ are sampled as before:

$$p((x, y)_\Gamma|do(a)) = \prod_{j \in \Gamma} p((x, y)_j|(x, y)_{pa(j)\cap\Gamma}, \theta_{pa(j)\cap I_a})$$
$$\text{with} \quad \Gamma := D \backslash I_a$$

Sampling from the subpopulation-based post-recourse distribution using $\mathcal{G}$ We recall that for actions a that potentially affect Y the subpopulation-based post-recourse distribution is defined as

$$P(Y^{post}, X^{post}|do(a), X_{G_a}^{post} = x_{G_a}^{pre}). \tag{6}$$

As we will see, the previously described sampling procedure can be applied. Therefore we apply the second rule of do-calculus to show that in Equation 6 conditioning on x_{G_a} is equal to intervening $do(X_{G_a} = x_{G_a})$. More specifically, if we remove all outgoing edges from X_{G_a} and all incoming edges to I_a, then X_{G_a} and X_Γ with $\Gamma := D \backslash I_a \cap G_a = d(I_a)$ are d-separated, meaning that conditioning and intervention are equivalent (Figure 3).

$$P((Y, X)_\Gamma^{post}|do(a), X_{G_a}^{post} = x_{G_a}^{pre})$$
$$= P((Y, X)_\Gamma^{post}|do(a), do(X_{G_a}^{post} = x_{G_a}^{pre}))$$

As follows we can leverage the procedure to sample interventional distributions to sample from the subpopulation-based post-recourse distribution. The procedure is illustrated in Algorithm 3.

B.3.1 Learning the conditional distributions $P(X_j|x_{pa(j)})$

In this work we assume that we have prior knowledge that allows us to sample from the components of the factorization ($P(X_j|x_{pa(j)})$, e.g. available if we know the SCM).

If the conditional distributions are not known, they can be learned from observational data; depending on which assumptions about distribution and functional can be made, different techniques may be employed. For categorical variables the problem reduces to standard supervised learning with cross-entropy loss. For linear Gaussian data, the conditional distribution can be estimated analytically from the covariance matrix [Page Jr, 1984]. A variety of estimation techniques exist for continuous settings with nonlinearities [Bishop, 1994, Bashtannyk and Hyndman, 2001, Sohn et al., 2015, Trippe and Turner, 2018, Winkler et al., 2019, Hothorn and Zeileis, 2021].

B.4 Optimization

Like the optimization problems for CE [Wachter et al., 2017, Tsirtsis and Gomez Rodriguez, 2020] or CR [Karimi et al., 2020b], the optimization problem for ICR is computationally challenging. It can be seen as a two-stage problem, where in the first stage the intervention targets I_a, and in the second stage the corresponding intervention values θ_a are optimized [Karimi et al., 2020b]. For the selection of intervention targets I_a alone $2^{d'}$ combinations exist, with $d' \leq d$ being the number of causes of Y. We jointly optimize the intervention targets and the intervention values using a genetic algorithm called NSGA-II [Deb et al., 2002]. For mixed categorical and continuous data, previous work in the field [Dandl et al., 2020] suggests to use NSGA-II in combination with *mixed integer evaluation strategies* [Li et al., 2013]. The exact hyperparameter configurations are reported in C.3.

B.5 Estimation of the optimal observational predictor h^* using the SCM

Instead of leveraging supervised learning with cross-entropy loss, we can factorize the optimal observational predictor as shown in Proposition 6 and then leverage the SCM for the estimation.

Proposition 6. *The optimal observational predictor can be factorized into conditional distributions of nodes given their parents (using the Markov factorization). More specifically, we yield*

$$p(y|x) = \frac{p(x,y)}{p(x)} = \frac{p(x,y)}{\sum_{y' \in \{0,1\}} p(x,y)} \tag{7}$$

$$\overset{\text{M.f.}}{=} \frac{p(y|x_{pa(j)}) \prod_{j \in D} p(x_j|(x,y)_{pa(j)})}{\sum_{y' \in \{0,1\}} p(y'|x_{pa(j)}) \prod_{j \in D} p(x_j|(x,y')_{pa(j)})} \tag{8}$$

$$= \frac{p(y|x_{pa(j)}) \prod_{j \in ch(y)} p(x_j|x_{pa(j)}, y)}{\sum_{y' \in \{0,1\}} p(y'|x_{pa(j)}) \prod_{j \in ch(y)} p(x_j|x_{pa(j)}, y')}. \tag{9}$$

It remains to show how the conditional distribution $p(x_j|x_{pa(j)})$ of a node given its parents can be estimated. Generally it holds that

$$p(x_j|x_{pa(j)}) \tag{10}$$

$$\overset{\text{law tot. prob.}}{=} \int_{\mathcal{U}_j} p(x_j|x_{pa(j)}, u_j)p(u_j|x_{pa(j)})du \tag{11}$$

$$\overset{\text{SCM, } u_j \perp x_{pa(j)}}{=} \int_{\mathcal{U}_j} [f(x_{pa(j)}, u_j) = x_j]p(u_j)du. \tag{12}$$

The integral can be approximated using Monte Carlo integration: we can sample from $p(u_j)$, compute the respective $\tilde{x}_j = f_j(x_{pa(j)}, \tilde{u}_j)$ and compute the proportion of cases where $x_j = \tilde{x}_j$. If X_j and U_j are continuous, this may require huge sample sizes to converge.

Furthermore, we may be able to leverage assumptions about f_j to derive a closed form solution. If f_j is invertible, the integral reduces to $p(x_j|x_{pa(j)}) = p(U_j = f_j^{-1}(x_j, x_{pa(j)}))$. For binary nodes with $x_j := [\sigma(l(x_{pa(j)})) \leq u_j]$ and $U_j \sim Unif(0,1)$, we directly see that $p(x_j|x_{pa(j)}) = \sigma(l(x_{pa(j)}))$.

C Details on Experiments

In this section we provide additional details on the experiments. More specifically, we explain which open-source libraries we use, how to access our code and how to reproduce the results in C.1. We formally introduce the synthetic and semi-synthetic datasets that we used in our experiments in C.2 and the corresponding figures. Details on hyperparameters, models as well as detailed results are reported in C.3 and the corresponding tables.

C.1 Implementation

The code relies of efficient tensor calculations with `numpy` [Harris et al., 2020], `pytorch` [Paszke et al., 2019] and `jax` [Bradbury et al., 2018]. For named dataframes we use `pandas` [pandas development team, 2020]. For plotting we rely on `matplotlib` [Hunter, 2007] and `seaborn` [Waskom, 2021]. We use the evolutionary optimization library deap [Fortin et al., 2012] and NSGA-II [Deb et al., 2002] to solve the combinatorial optimization problem.[21] In order to speed up the computation, we cache queries and results for the improvement confidence using `functools.cache`. For continuous variables the intervention can be rounded to a specified number of digits to increase the probability of reusing a cached result (with neglectable loss of precision).[22]

All code is publicly available via `https://github.com/gcskoenig/icr`. The repository contains the user-friendly python package `icr`, which we use in our experiments to generate and evaluate recourse. Furthermore, the scripts for the experiments, the scripts for the visualization of the results as well as a `README.md` with instructions for the installation of all dependencies are contained in the repository, such that the experiments are reproducible.

C.2 Synthetic and Semi-Synthetic Datasets

3var-causal and *3var-noncausal* are abstract, synthetic settings. *5var-skill* is inspired by Montandon et al. [2021], who use GitHub profiles to detect the role of a developer. In our SCM we model *senior-level skill* as a binary variable which is caused by *programming experience* and the education *degree*. The skill is causal for GitHub metrics such as the number of *commits*, the number of programming *languages* and the number of *stars*. The *7var-covid* dataset is inspired by Jehi et al. [2020]. The following variables are introduced: population density D, flu vaccination V_I, number of covid vaccination shots V_C, deviation from average BMI B, whether someone is free of covid disease C, whether the individual has influence I, appetite loss S_A, fever S_{Fe} and fatigue S_{Fa}. The corresponding structural equations, noise distributions and causal graphs are provided in Figure 4 (*3var-causal*), 5 (*3var-noncausal*), 6 (*5var-skill*) and 7 (*7var-covid*). A pairplot for each dataset is presented in Figure 8. In our notation σ is the sigmoid function, N the Gaussian distribution, Cat a categorical distribution, $Unif$ the uniform distribution, $Bern$ a Bernoulli distribution and GaP a Gamma-Poisson mixture. $[cond]$ is 1 when the condition is met and 0 if not. As a consequence variables with $[Z \leq U]$ and $U \sim Unif(0, 1)$ are bernoulli distributed with $Bern(Z)$.

C.3 Detailed Results

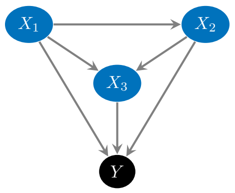
In this section we report all experimental results in tabular form. More specifically, the results for *3var-causal* are reported in Table 2, for *3var-noncausal* in Table 3, for *5var-skill* in Table 4 and for *7var-covid* in Table 5. For each experiment we report the specified confidence γ (or η for CR), as well as the observed improvement rate γ_{obs}, the observed acceptance rate η_{obs}, the observed acceptance rate by the individualized post-recourse predictor $\eta_{obs}^{\text{indiv.}}$, the observed acceptance rate on refits $\eta_{obs}^{\text{refit}}$ and the average recourse cost for individuals who were rejected and whom were provided with a recourse recommendation. A visual summary of the results is provided in Section 8.

In order to enable a more direct comparison of the CR and ICR targets, we equalize the optimization thresholds for ICR and CR. More specifically, for CR we require the (individualized or subpopulation-based) acceptance probability to be $\geq \eta$, and for ICR we require the (individualized or subpopulation-based) improvement probability to be $\geq \overline{\gamma}$, where $\overline{\gamma} = \overline{\eta}$.[23] Furthermore, in order to be able to estimate the effects of recourse actions, CR assumes causal sufficiency, meaning that there are no two endogeneous variables that share an unobserved cause. If the target variable Y is exogeneous then any causal model with more than one endogeneous direct effect of Y violates the assumptions. In order to enable an application of CR on datasets with more than one effect variable we assume knowledge of the SCM

[21] We also implemented abduction based on probabilistic inference. Thereby we rely on on `pyro` [Bingham et al., 2018] for discrete inference and `numpyro` [Phan et al., 2019] for MCMC inference of continuous variables. For our experiments we used the analytical formulas presented in B

[22] All packages are open source. For detailed license information we refer to the respective package websites.

[23] A short comment on the choice of a non-adaptive threshold can be found in E.2.

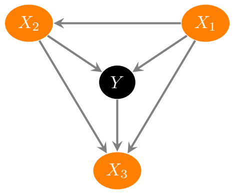


$$\begin{aligned} X_1 &:= U_1, & U_1 &\sim N(0,1) \\ X_2 &:= X_1 + U_2, & U_2 &\sim N(0,1) \\ X_3 &:= X_1 + X_2 + U_3, & U_3 &\sim N(0,1) \\ Y &\sim [\sigma(X_1 + X_2 + X_3) \leq U_Y], & U_Y &\sim Unif(0,1) \end{aligned}$$

(a) Causal graph

(b) Structural Equations

Figure 4: SCM for *3var-causal*. The cost function is given as $cost(a) = \delta_1 + \delta_2 + \delta_3$, where δ is the vector of absolute changes to the intervened upon variables. E.g., for $do(a) = do(X_1 = x_1')$, $\delta_1 = |x_1' - x_1|$ and $\delta_2 = \delta_3 = 0$

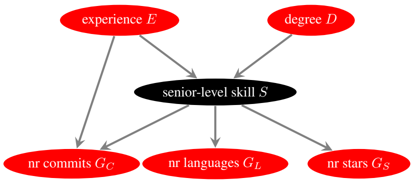


$$\begin{aligned} X_1 &:= U_1, & U_1 &\sim N(0,1) \\ X_2 &:= X_1 + U_1, & U_1 &\sim N(0,1) \\ Y &:= [\sigma(X_1 + X_2) \leq U_Y], & U_Y &\sim Unif(0,1) \\ X_3 &:= X_1 + X_2 + Y + U_3, & U_3 &\sim N(0,0.1) \end{aligned}$$

(a) Causal graph

(b) Structural Equations

Figure 5: SCM for *3var-noncausal* with $cost(a) = \delta_1 + \delta_2 + \delta_3$.

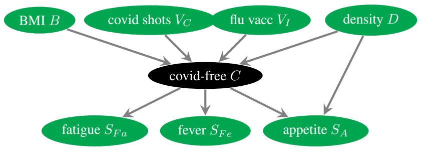


$$\begin{aligned} E &:= U_E; U_E \sim GaP(8, 8/3) \\ D &:= U_D; U_D \sim Cat(0.4, 0.2, 0.3, 0.1) \\ S &:= [\sigma(-10 + 3E + 4D)) \leq U_S]\,; U_S \sim Unif(0,1) \\ G_C &:= 10E(11 + 100D) + U_{G_C}; U_{G_C} \sim GaP(40, 40/4) \\ G_L &:= \sigma(10S) + U_{G_L}; U_{G_L} \sim GaP(2, 2/4) \\ G_S &:= 10S + U_{G_S}; U_{G_S} \sim GaP(5, 5/4) \end{aligned}$$

(a) Causal graph

(b) Structural Equations

Figure 6: SCM for *5var-skill* with $cost(a) = 5\delta_E + 5\delta_D + 0.0001\delta_{G_C} + 0.01\delta_{G_L} + 0.1\delta G_S$.

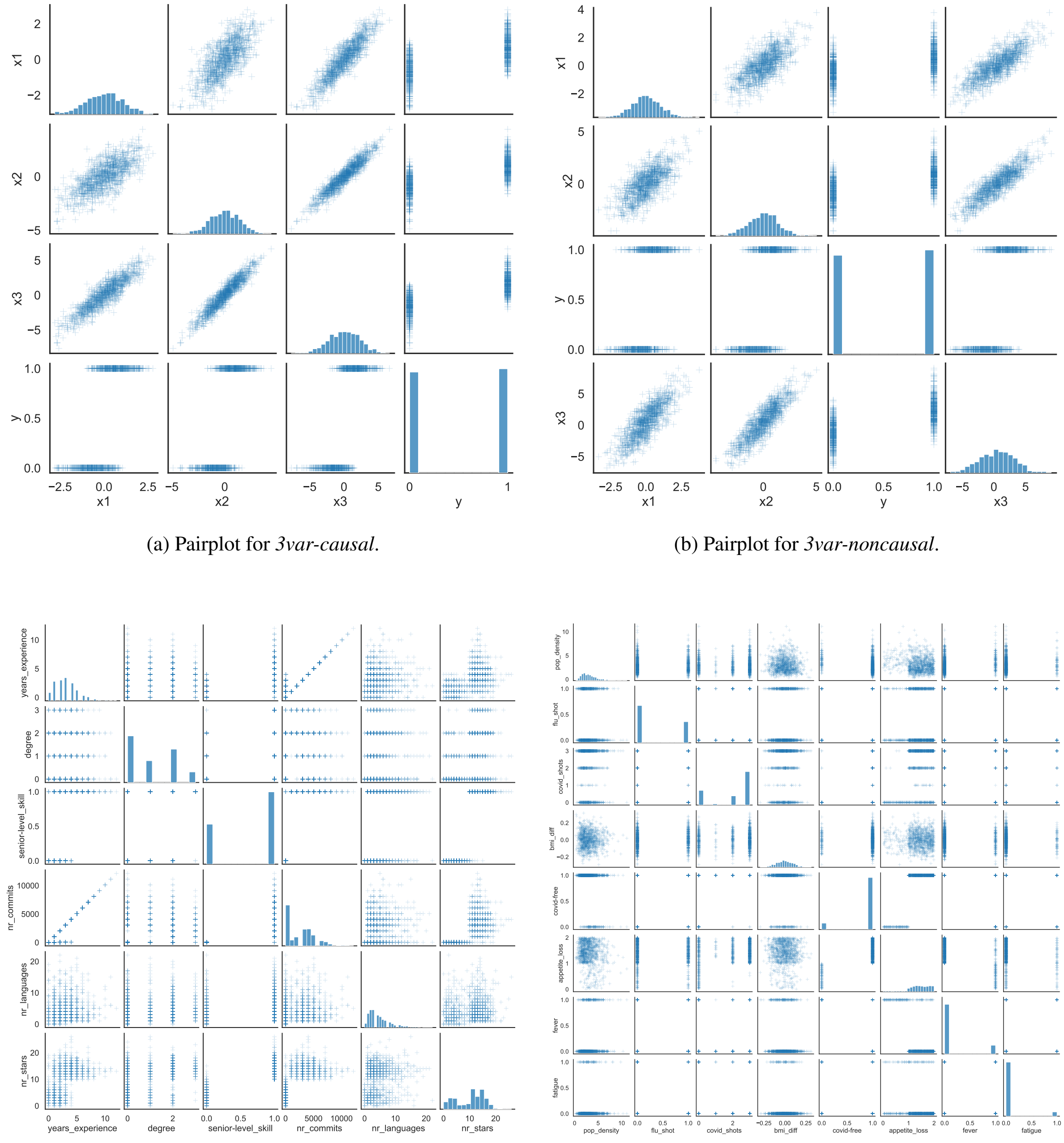


$$\begin{aligned} D &:= U_D; U_D \sim \Gamma(4, 4/3) \\ V_I &:= U_{V_I}; U_{V_I} \sim Bern(0.39) \\ V_C &:= U_{V_C}; U_{V_C} \sim Cat(0.24, 0.02, 0.15, 0.59) \\ B &:= U_B; U_B \sim N(0,1) \\ C &:= \left[\sigma(-(D - 3 - V_I - 2.5V_C + 0.2B^2)) \leq U_C\right]; \\ U_C &\sim Unif(0,1) \\ S_A &:= [\sigma(-2C) \leq U_{S_A}]\,; U_{S_A} \sim Unif(0,1) \\ S_{Fe} &:= [\sigma(5 - 9C) \leq U_{S_{Fe}}]\,; U_{S_{Fe}} \sim Unif(0,1) \\ S_{Fa} &:= \left[\sigma(-1 + B^2 - 2C) \leq U_{S_{Fa}}\right]; \\ U_{S_{Fa}} &\sim Unif(0,1) \end{aligned}$$

(a) Causal graph

(b) Structural Equations

Figure 7: SCM for *7var-covid* with cost function $cost(a) = \delta_D + \delta_{V_I} + \delta_{V_C} + \delta_B + \delta_{S_A} + \delta_{S_{Fe}} + \delta_{S_{Fa}}$.



(a) Pairplot for *3var-causal*.

(b) Pairplot for *3var-noncausal*.

(c) Pairplot for *5var-skill*.

(d) Pairplot for *7var-covid*.

Figure 8: Pairplots for the SCMs.

Table 2: Results for 3var-causal.

3var-causal	$\overline{\gamma}$ / $\overline{\eta}$	$\gamma_{\text{obs.}}$	±	$\eta_{\text{obs.}}$	±	$\eta^{individ.}_{\text{obs.}}$	±	$\eta^{\text{refit}}_{\text{obs.}}$	±	∅ cost	±
CE	-	0.41	0.09	1.00	0.00	-	-	0.60	0.20	3.08	0.41
ind. CR	0.75	0.47	0.10	1.00	0.00	-	-	0.70	0.10	2.46	0.37
ind. CR	0.85	0.44	0.08	1.00	0.00	-	-	0.72	0.12	2.39	0.25
ind. CR	0.90	0.47	0.09	1.00	0.00	-	-	0.72	0.14	2.36	0.35
ind. CR	0.95	0.49	0.07	1.00	0.00	-	-	0.67	0.10	2.44	0.31
subp. CR	0.75	0.46	0.11	0.86	0.04	-	-	0.64	0.14	2.66	0.41
subp. CR	0.85	0.43	0.08	0.93	0.02	-	-	0.69	0.14	2.64	0.32
subp. CR	0.90	0.45	0.09	0.96	0.02	-	-	0.70	0.15	2.73	0.42
subp. CR	0.95	0.48	0.09	0.98	0.01	-	-	0.64	0.14	2.86	0.41
ind. ICR	0.75	0.79	0.06	0.98	0.02	1.0	0.0	0.96	0.03	3.27	0.50
ind. ICR	0.85	0.86	0.03	1.00	0.01	1.0	0.0	0.97	0.02	3.82	0.30
ind. ICR	0.90	0.90	0.02	1.00	0.01	1.0	0.0	0.98	0.03	3.70	0.31
ind. ICR	0.95	0.95	0.01	1.00	0.00	1.0	0.0	0.99	0.01	4.08	0.24
subp. ICR	0.75	0.75	0.04	0.93	0.04	-	-	0.90	0.04	3.34	0.49
subp. ICR	0.85	0.87	0.03	0.98	0.01	-	-	0.96	0.02	4.05	0.29
subp. ICR	0.90	0.89	0.02	0.99	0.01	-	-	0.97	0.02	3.87	0.25
subp. ICR	0.95	0.94	0.02	1.00	0.00	-	-	0.99	0.01	4.22	0.28

Table 3: Results for 3var-noncausal

3var-noncausal	$\overline{\gamma}$ / $\overline{\eta}$	$\gamma_{\text{obs.}}$	±	$\eta_{\text{obs.}}$	±	$\eta^{individ.}_{\text{obs.}}$	±	$\eta^{\text{refit}}_{\text{obs.}}$	±	∅ cost	±
CE	-	0.17	0.03	0.98	0.04	-	-	0.67	0.15	2.28	0.26
ind. CR	0.75	0.25	0.03	1.00	0.00	-	-	0.70	0.13	2.28	0.21
ind. CR	0.85	0.24	0.02	1.00	0.00	-	-	0.73	0.13	2.29	0.17
ind. CR	0.90	0.24	0.04	1.00	0.00	-	-	0.71	0.11	2.24	0.16
ind. CR	0.95	0.23	0.04	1.00	0.00	-	-	0.73	0.12	2.18	0.32
subp. CR	0.75	0.22	0.03	0.91	0.03	-	-	0.63	0.15	2.18	0.12
subp. CR	0.85	0.19	0.03	0.95	0.02	-	-	0.67	0.15	2.33	0.21
subp. CR	0.90	0.19	0.03	0.97	0.01	-	-	0.65	0.14	2.42	0.19
subp. CR	0.95	0.19	0.03	0.99	0.01	-	-	0.69	0.14	2.26	0.32
ind. ICR	0.75	0.77	0.03	0.93	0.02	0.79	0.03	0.93	0.02	2.16	0.11
ind. ICR	0.85	0.86	0.02	0.99	0.01	0.90	0.02	0.99	0.01	2.51	0.08
ind. ICR	0.90	0.91	0.03	1.00	0.00	0.94	0.01	1.00	0.00	3.00	0.08
ind. ICR	0.95	0.96	0.02	0.98	0.07	0.98	0.01	0.98	0.08	3.32	0.16
subp. ICR	0.75	0.69	0.03	0.77	0.05	-	-	0.76	0.05	2.11	0.20
subp. ICR	0.85	0.82	0.03	0.93	0.02	-	-	0.92	0.02	2.42	0.11
subp. ICR	0.90	0.89	0.03	0.98	0.01	-	-	0.97	0.01	2.86	0.13
subp. ICR	0.95	0.94	0.02	0.97	0.10	-	-	0.96	0.12	3.19	0.15

Table 4: Results for 5var-skill

5var-skill	$\overline{\gamma}$ / $\overline{\eta}$	$\gamma_{\text{obs.}}$	±	$\eta_{\text{obs.}}$	±	$\eta_{\text{obs.}}^{individ.}$	±	$\eta_{\text{obs.}}^{\text{refit}}$	±	$\emptyset$ cost	±
CE	-	0.00	0.00	1.00	0.00	-	-	0.76	0.14	1.34	1.28
ind. CR	0.75	0.00	0.00	1.00	0.00	-	-	0.86	0.11	0.27	0.28
ind. CR	0.85	0.00	0.00	1.00	0.00	-	-	0.81	0.14	0.24	0.20
ind. CR	0.90	0.00	0.01	1.00	0.00	-	-	0.70	0.15	0.10	0.00
ind. CR	0.95	0.00	0.00	1.00	0.00	-	-	0.66	0.16	0.11	0.03
subp. CR	0.75	0.00	0.00	1.00	0.00	-	-	0.85	0.11	4.06	4.97
subp. CR	0.85	0.00	0.00	1.00	0.00	-	-	0.80	0.15	0.24	0.19
subp. CR	0.90	0.00	0.01	1.00	0.00	-	-	0.70	0.15	0.10	0.01
subp. CR	0.95	0.00	0.00	1.00	0.00	-	-	0.66	0.15	0.12	0.04
ind. ICR	0.75	0.94	0.02	0.94	0.02	0.94	0.02	0.94	0.02	4.95	5.32
ind. ICR	0.85	0.94	0.01	0.93	0.02	0.94	0.01	0.93	0.02	9.80	0.27
ind. ICR	0.90	0.96	0.02	0.96	0.02	0.96	0.02	0.96	0.02	10.38	0.23
ind. ICR	0.95	0.98	0.01	0.98	0.01	0.98	0.01	0.98	0.01	11.23	0.21
subp. ICR	0.75	0.93	0.01	0.93	0.02	-	-	0.93	0.01	4.72	5.08
subp. ICR	0.85	0.94	0.01	0.94	0.01	-	-	0.94	0.02	9.74	0.17
subp. ICR	0.90	0.96	0.01	0.96	0.01	-	-	0.96	0.01	10.46	0.53
subp. ICR	0.95	0.97	0.01	0.97	0.01	-	-	0.97	0.01	10.88	0.21

Table 5: Results for 7var-covid

7var-covid	$\overline{\gamma}$ / $\overline{\eta}$	$\gamma_{\text{obs.}}$	±	$\eta_{\text{obs.}}$	±	$\eta_{\text{obs.}}^{individ.}$	±	$\eta_{\text{obs.}}^{\text{refit}}$	±	$\emptyset$ cost	±
CE	-	0.00	0.00	1.00	0.00	-	-	1.00	0.00	0.60	0.12
ind. CR	0.75	0.01	0.00	1.00	0.00	-	-	0.99	0.01	0.56	0.02
ind. CR	0.85	0.00	0.00	1.00	0.00	-	-	0.99	0.00	0.55	0.02
ind. CR	0.90	0.00	0.00	1.00	0.00	-	-	1.00	0.00	0.55	0.03
ind. CR	0.95	0.00	0.00	1.00	0.00	-	-	0.99	0.01	0.54	0.07
subp. CR	0.75	0.01	0.01	0.92	0.02	-	-	0.91	0.02	0.52	0.03
subp. CR	0.85	0.00	0.01	0.97	0.01	-	-	0.96	0.01	0.75	0.40
subp. CR	0.90	0.00	0.00	0.98	0.01	-	-	0.98	0.01	0.55	0.03
subp. CR	0.95	0.00	0.00	0.99	0.01	-	-	0.98	0.01	0.51	0.07
ind. ICR	0.75	0.81	0.03	0.81	0.03	0.82	0.04	0.81	0.03	1.26	0.02
ind. ICR	0.85	0.85	0.03	0.85	0.03	0.86	0.03	0.85	0.03	1.14	0.44
ind. ICR	0.90	0.89	0.03	0.89	0.03	0.90	0.02	0.89	0.03	1.61	0.02
ind. ICR	0.95	0.95	0.01	0.95	0.01	0.95	0.01	0.95	0.01	1.97	0.06
subp. ICR	0.75	0.61	0.04	0.61	0.04	-	-	0.61	0.04	1.06	0.03
subp. ICR	0.85	0.73	0.03	0.73	0.03	-	-	0.73	0.03	1.09	0.34
subp. ICR	0.90	0.81	0.04	0.81	0.04	-	-	0.81	0.04	1.42	0.05
subp. ICR	0.95	0.90	0.03	0.90	0.03	-	-	0.90	0.03	1.73	0.06

including Y for CR as well and draw ground-truth interventional samples from the SCM instead of identifying the interventional distribution from observational data.

For *3var-causal* and *3var-noncausal* we configured NSGA-II to optimize over 600 generations with a population size of 300, for *5var-skill* and *7var-covid* 1000 generations with 500 individuals were used. For all experiments the crossover probability was 0.3 and the mutation probability 0.05. For all settings continuous variables were rounded to 1 decimal point. For the 3 variable settings a standard `sklearn` `LogisticRegression` was used, for the refits without penality. For the nonlinear dataset a `RandomForestClassifier` with max depth 30, 50 estimators and balanced subsampling was applied. The experimental results were computed on a Quad core Intel Core i7-7700 Kaby Lake processor. For each setting, the experiments took between 24 to 48 hours.

D Proofs

As follows we provide the full proofs for Propositions 1 - 5.

D.1 Linking individualized prediction with γ^{ind}, Proof of Proposition 1

Proposition 1. *The expected individualized post-recourse score is equal to the individualized improvement probability $\gamma^{ind}(x^{pre}, a) := P(Y^{post} = 1|x^{pre}, do(a))$, i.e.*

$$E[\hat{h}^{*,ind}(x^{post})|x^{pre}, do(a)] = \gamma^{ind}(a).$$

Proof: It holds that

$$\begin{aligned}&E[h^{*,ind}(x^{post})|x^{pre}, do(a)]\\&= E[E[Y|x^{pre}, x^{post}]|x^{pre}, do(a)]\\&\overset{\text{total exp.}}{=} E[Y|x^{pre}, do(a)]\\&= \gamma^{ind}(a).\end{aligned}$$

D.2 Intervention stability w.r.t. ICR actions, Proposition 2

Proposition 2. *Given nonzero cost for all interventions, ICR exclusively suggests actions on causes of Y. Assuming causal sufficiency, for any optimal predictor the conditional distribution of Y given the variables that the model uses X_S (i.e. $P(Y|X_S)$) is stable w.r.t interventions on causes. Therefore, optimal predictors are intervention stable w.r.t. ICR actions.*

Proof: We prove the statement in six steps.

ICR only intervenes on causes: The goal of meaningful recourse is to improve Y with minimal cost. Only interventions on causes alter Y. Consequently, actions on non-causes of Y would not be suggested by meaningful recourse.

Given causal sufficiency, a graph $\mathcal{G}$ and an endogenous Y, the set of endogeneous direct parents, direct effects and direct parents of effects are the minimal d-separating set $S_{\mathcal{G}}$: Standard result, see e.g. Peters et al. [2017], Proposition 6.27.

The set $S_{\mathcal{G}^*}$ in the augmented graph $\mathcal{G}^*$ coincides with $S_{\mathcal{G}}$: The minimal d-separating set contains direct causes, direct effects and direct parents of direct effects. I_l is never a direct cause of X_l. Also, since I_l has no endogenous causes, it cannot be a direct effect. Furthermore, since we restrict interventions to be performed on causes, I_l cannot be a direct parent of a direct effect.

$S_{\mathcal{G}}$ is intervention stable: As follows, all intervention variables are d-separated from Y in $\mathcal{G}^*$ by $S_{\mathcal{G}}$. Therefore $S_{\mathcal{G}}$ is intervention stable. An example is given in Figure 9.

Then also the markov blanket is intervention stable: Since d-separation implies independence $MB(Y) \subseteq S_{\mathcal{G}}$. Therefore, if $X_T \perp Y|X_{MB(Y)}$ then also $X_T \perp Y|S_{\mathcal{G}}$. If any element $s \in S_{\mathcal{G}}$ it holds that $s \notin MB(Y)$, then it must hold that $X_s \perp Y|X_{MB(Y)}$. Therefore, if $X_T \perp Y|X_{MB(Y)}, X_s$ then also $X_T \perp Y|X_{MB(Y)}$ and therefore any independence entailed by $S_{\mathcal{G}}$ also holds for $MB(Y)$. Since Pfister et al. [2021] only require the independence that is implied by d-separation in their invariant conditional proof, the same implication holds for the $MB(Y)$. As follows, $P(Y|X_{MB(Y)})$ is invariant with respect to interventions on any set of endogenous causes.

Then any superset of the markov blanket is intervention stable: We prove the statement by contradiction. The markov blanket d-separates the target variable Y from any other set of variables. If adding a set of variables S_1 to the markov blanket would open a path to any other set of variables S_2, then it would hold that $S := S_1 \cup S_2$ is not d-separated from Y ($P(Y|MB(Y)) = P(Y|MB(Y), S_1, S_2) \neq P(Y|MB(Y), S_1) = P(Y|MB(Y))$)

D.3 Linking observational prediction and γ^{sub}, Proposition 3

Proposition 3. *Given causal sufficiency and positivity*[24]*, for interventions on causes the expected subgroup-wide optimal score h^* is equal to the subgroup-wide improvement probability $\gamma^{sub}(a) := P(Y^{post} = 1|do(a), x^{pre}_{G_a})$, i.e.*

$$E[\hat{h}^*(x^{post})|x^{pre}_{G_a}, do(a)] = \gamma^{sub}(a).$$

[24] Positivity ensures that the post-recourse observation lies within the observational support , where the model was trained (i.e., $p^{pre}(x^{post}) > 0$), [Neal, 2020]).

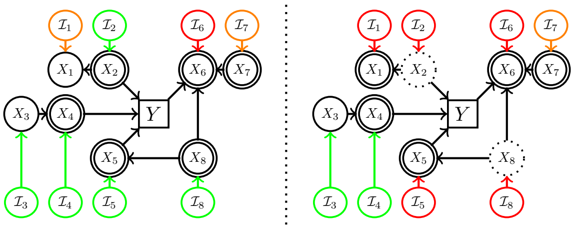


Figure 9: A schematic drawing illustrating under which interventions $I_1, \ldots, I_8$ the Markov blanket (double circle) is intervention stable. In this setting, we consider the intervention variables to be independent treatment variables: We would like to know how the different actions influence the conditional distribution, irrespective of how likely they are to be applied. Therefore, they are modeled as parent-less variables. Green indicates intervention stability, red indicates no intervention stability. Orange indicates intervention stability of non-causal variables. Dotted variables are not observed. *Left:* Since all endogenous variables are observed, $MB_O(Y)$ is stable w.r.t. interventions on every endogenous cause of Y (Proposition 3). *Right:* Unobserved variables (X_2, X_8) open paths between interventions on causes and Y.

Proof: The proposition follows from Proposition 2. More specifically

$$E[h^*(x^{post,a})|x_G^{pre},a] = E[E[Y|x^{post,a}]|x_G^{pre},a] \overset{\text{total exp.}}{=} E[Y|x_G^{pre},a] \overset{\text{def. }\gamma^{sub}}{=} \gamma^{sub}(a). \tag{13}$$

D.4 Acceptance Bound, Proof of Proposition 4

Proposition 4. *Let g be a predictor with $E[g(x^{post})|x_S^{pre}, do(a)] = \gamma(x_S^{pre}, a)$. Then for a decision threshold t the post-recourse acceptance probability $\eta(t; x_S^{pre}, a) := P(g(x^{post}) > t|x_S^{pre}, do(a))$ is lower bounded:*

$$\eta(t; x_S^{pre}, a) \geq \frac{\gamma(x_S^{pre}, a) - t}{1 - t}.$$

Proof: Positivity ($p^{pre}(x^{post}) > 0$) is necessary for subpopulation-based ICR since only then we can assume that the model is actually optimal for any input that it receives. The problem is discussed in more detail in Hernán MA [2020], Neal [2020].

As follows we denote $\hat{h}^*$ as the random variable indicating the predictions of the post-recourse predictors described in Section 5.
From Propositions 1 and 3, for both individualized and subpopulation-based post-recourse predictors we know that

$$E[\hat{h}(x^{post,a})^*|x_S^{pre}, do(a)] = \gamma(x_S^{pre}, a).$$

We decompose the expected prediction

$$\begin{aligned}
\gamma(x_S^{pre}, a) &= E[\hat{h}^*|x_S^{pre}, a] && (14)\\
&= E[\hat{h}^*|\hat{h}^* > t]P(\hat{h}^* > t) + E[\hat{h}^*|\hat{h}^* \leq t]P(\hat{h}^* \leq t)\Big|_{x_S^{pre},a} && (15)\\
&= E[\hat{h}^*|\hat{h}^* > t]P(\hat{h}^* > t) + E[\hat{h}^*|\hat{h}^* \leq t](1 - P(\hat{h}^* > t))\Big|_{x_S^{pre},a} && (16)\\
&= E[\hat{h}^*|\hat{h}^* > t]P(\hat{h}^* > t) + E[\hat{h}^*|\hat{h}^* \leq t] - P(\hat{h}^* > t)E[\hat{h}^*|\hat{h}^* \leq t]\Big|_{x_S^{pre},a} && (17)\\
&= E[\hat{h}^*|\hat{h}^* \leq t] + P(\hat{h}^* > t)\Big(E[\hat{h}^*|\hat{h}^* > t] - E[\hat{h}^*|\hat{h}^* \leq t]\Big)\Big|_{x_S^{pre},a} && (18)
\end{aligned}$$

which can be reformulated to yield the acceptance rate η:

$$\frac{\gamma - E[\hat{h}^*|\hat{h}^* \leq t]}{E[\hat{h}^*|\hat{h}^* > t] - E[\hat{h}^*|\hat{h}^* \leq t]}\Bigg|_{x_S^{pre},a} = P(\hat{h}^* > t|x_S^{pre}, a) = \eta(x_S^{pre}, a). \tag{19}$$

It holds that $E[\hat{h}^{*,ind}|\hat{h}^* \leq t] = FNR(t)$ and $E[\hat{h}^*|\hat{h}^* > t] = TPR(t)$.

We can show that $E[\hat{h}^*|\hat{h}^* \leq t] \leq t$:

$$\begin{aligned}0 &\leq FNR(t|x_S^{pre}, a) && (20)\\ &= P(Y^{a,post} = 1|h^* \leq t, x_S^{pre}, a) && (21)\\ &= E[Y^{a,post}|h^* \leq t, x_S^{pre}, a] && (22)\\ &= E[E[Y^{a,post}|x^{post,a}]|h^* \leq t, x_S^{pre}, a] && (23)\\ &= E[h^*|h^* \leq t, x_S^{pre}, a] && (24)\\ &\leq t && (25)\end{aligned}$$

and analog that $1 \geq TPR(t) \geq t$. Therefore

$$\eta(t, x_S^{pre}, a) = \frac{\gamma - FNR(t)}{TPR(t) - FNR(t)}\bigg|_{x_S^{pre}, a} \geq \frac{\gamma(x_S^{pre}, a) - FNR(t)}{1 - FNR(t)} \geq \frac{\gamma(x_S^{pre}, a) - t}{1 - t}. \quad (26)$$

D.5 Individualized post-recourse prediction, proof of Proposition 5

Proposition 5. *In general, the individualized post-recourse predictor can be estimated as*

$$\begin{aligned}&p(y^{post}|x^{pre}, x^{post}, do(a)) && (27)\\ &= \frac{\int_{\mathcal{U}} p(y^{post}, x^{post}|u, do(a))p(u|x^{pre})du}{\sum_{y' \in \{0,1\}} \left(\int_{\mathcal{U}} p(y', x^{post}|u, do(a))p(u|x^{pre})du\right)} && (28)\end{aligned}$$

Given binary decision problems with invertible structural equations, the individualized post-recourse prediction function reduces to

$$\begin{aligned}&p(y^{post}|x^{post}, x^{pre}, do(a)) && (29)\\ &= \frac{p(U_{-I} = f^{-1}_{do(a)}(y^{post}, x^{post})|x^{pre}, do(a))}{\sum_{y' \in \{0,1\}} p(U_{-I} = f^{-1}_{do(a)}(y', x^{post})|x^{pre}, do(a))}. && (30)\end{aligned}$$

Proof: It holds that

$$p(y^{post}|x^{pre}, x^{post}, do(a)) \overset{\text{def. cond.}}{=} \frac{p(y^{post}, x^{post}|x^{pre}, do(a))}{p(x^{post}|x^{pre}, do(a))} \quad (31)$$

$$(32)$$

We can reformulate the conditional distribution $p(y^{post}, x^{post}|x^{pre}, do(a))$ as two parts, one that describes the probability of a state of the context given x^{pre}, and one that describes the probability of a post-recourse state x^{post}, y^{post} given a certain noise state u and $do(a)$.

$$\begin{aligned}&p(y^{post}, x^{post}|x^{pre}, do(a)) && (33)\\ &\overset{\text{marginal.}}{=} \int_{\mathcal{U}} p(y^{post}, x^{post}, u|x^{pre}, do(a))du && (34)\\ &\overset{\text{chain rule}}{=} \int_{\mathcal{U}} p(y^{post}, x^{post}|u, x^{pre}, do(a))p(u|x^{pre})du && (35)\\ &\overset{(y,x)^{post} \perp x^{pre}|u}{=} \int_{\mathcal{U}} p(y^{post}, x^{post}|u, do(a))p(u|x^{pre})du. && (36)\end{aligned}$$

In combination we yield

$$\begin{aligned}&p(y^{post}|x^{pre}, x^{post}, do(a)) && (37)\\ &= \frac{\int_{\mathcal{U}} p(y^{post}, x^{post}|u, do(a))p(u|x^{pre})du}{\int_{\mathcal{Y}} \left(\int_{\mathcal{U}} p(y', x^{post}|u, do(a))p(u|x^{pre})du\right) dy'} && (38)\\ &= \frac{\int_{\mathcal{U}} p(y^{post}, x^{post}|u, do(a))p(u|x^{pre})du}{\sum_{y' \in 0,1} \left(\int_{\mathcal{U}} p(y', x^{post}|u, do(a))p(u|x^{pre})du\right)} && (39)\end{aligned}$$

For a setting with invertible structural equations this reduces to

$$
\begin{aligned}
&p(y^{post}|x^{post}, x^{pre}, do(a)) && (40)\\
&= \frac{p(y^{post}, x^{post}|x^{pre}, do(a))}{p(x^{post}|x^{pre}, do(a))} && (41)\\
&= \frac{p(U_{-I} = f^{-1}(y^{post}, x^{post})|x^{pre}, do(a))}{\sum_{y' \in \{0,1\}} p(U_{-I} = f^{-1}(y^{post}, x^{post})|x^{pre}, do(a))}. && (42)
\end{aligned}
$$

where $-I$ is the index set for variables that have not been intervened on (since the noise terms for the intervened upon variables are isolated variables in the interventional graph).

E Misc

E.1 Negative Result: Algorithmic recourse is neither meaningful nor robust

In the introduction we claimed that CR recommendations [Karimi et al., 2020b, 2021] may not lead to improvement. Now, we formally demonstrate the case on the Covid hospital admission example (Figure 1) which we extend with the full structural causal model (Example 1). Furthermore, we show that CR is not robust to refits of the model on mixed pre- and post-recourse data. All code is publicly available via `https://anonymous.4open.science/r/icr-aaai/README.md`.

Example 1. *Let V indicate whether someone is fully vaccinated, Y indicate whether someone is free of Covid and S whether someone is asymptomatic. The data is generated by the following structural causal model (SCM) entailing the causal graph depicted in Figure 1:*

$$V := U_V, \qquad U_V \sim Bern(0.5) \tag{43}$$

$$Y := V + U_Y \ \mathtt{mod}\ 2, \qquad U_Y \sim Bern(0.09) \tag{44}$$

$$S := Y + U_S \ \mathtt{mod}\ 2, \qquad U_S \sim Bern(0.05) \tag{45}$$

For prediction, a `sklearn` *logistic regression model is fit on* 2000 *samples, yielding* $\hat{h}$ *with* $\beta_v \approx 3.7$*,* $\beta_s \approx 5.1$*,* $\beta_0 \approx -4.3$*. Visitors are allowed to enter the hospital if* $\hat{h} < 0.5$*. Intervening on (flipping)* V *and* S *costs* 0.5 *and* 0.1 *respectively.*

Lack of improvement: Given a decision threshold of 0.5, the model admits everyone without symptoms ($S = 1$), irrespective of their vaccination status V. Therefore, in order to revert rejections ($S = 0$), both individualized and subpopulation-based CR suggest removing the symptoms S ($do(S = 1)$, for instance by taking cough drops). However, since they only treat the symptoms S, the actual Covid risk Y is unaffected: none of the recourse-implementing individuals actually improve. We say the predictor is *gamed.*

Lack of robustness: For individuals who implement recourse the association between symptom state S and Covid risk Y is broken. Thus, the predictive power of the model for recourse-seeking individual drops from ≈ 95 percent pre-recourse to ≈ 5 percent post-recourse.[25] A refit of the model on a mix pre- and post-recourse data (2000 samples each) yields $\hat{h}$ with $\beta_V \approx 4.1, \beta_S \approx 3.3, \beta_0 \approx -4.8$. Since the association between symptom state and disease status is broken post-recourse, the new model rejects individuals if they are not vaccinated, irrespective of their symptom state. For that reason, recourse recommendations that were designed for the original model only lead to acceptance by the refitted model for those individuals who happened to be vaccinated anyway.

The example demonstrates that CR recommendations are prone to gaming the predictor and therefore may neither lead to improvement nor be robust to model refits.

E.2 Interpretability of improvement confidence γ

Counterfactuals are concerned with changing the inputs to the model such that the model prediction changes in the desired way. Since the prediction function is deterministic and accessible, the post-recourse prediction can be determined exactly.

In contrast CR and ICR deal with the effects of real-world interventions on real-world variables. As such, the effects of recourse actions on the covariates (and the underlying prediction target) cannot be determined exactly. Therefore both CR and ICR have to deal with uncertainty.

CR deals with this uncertainty by phrasing the optimization objective for CR in terms of an expectation over the prediction distribution and by using an action-adaptive confidence threshold. This threshold `thresh` bounds the expected prediction away from the model's decision threshold (e.g. $t = 0.5$). Using the conservativeness parameters, the user can roughly steer how far the expected prediction shall be away from the decision boundary.

In contrast, ICR deals with the uncertainty by letting the user specify the confidence γ, which can be intuitively interpreted as improvement probability (whereas the expected prediction cannot be interpreted as acceptance probability). A lower-bound on the acceptance probability for a combination of γ and t is given in Proposition 4. Furthermore, we can estimate the individualized and subpopulation-based acceptance rates for a specific situation (a, x^{pre}) as detailed in B.1 and B.3. The human-interpretable improvement and acceptance confidences are vital for the explainee to make an informed decision.

In order to allow a direct comparison of the methods, we rephrase the CR objective to optimize the acceptance probability η in our experiments.

[25] The previously wrongly-rejected individuals are correctly classified after implementing recourse.

E.3 Imbalance between standard predictors and individualized ICR recommendations

In Section 6 we argued that there is an imbalance in predictive capability between (optimal) observational predictors and the pre-recourse SCM (which used to predict γ^{ind}). We illustrate the problem on a simple example.

Example 2. *Let there be a three variable chain $X_1 \rightarrow Y \rightarrow X_2$ where at every step the value is incremented by one with 50% chance and the maximum value is set to 2 ($X_1 := U_1$, $Y := X_1 + U_Y$, $X_2 := min(2, Y + U_2)$ where $U_1, U_2, U_Y \sim Bern(0.5)$). Let us assume a factual observation $x^{pre} = (0, 2)$ and action $a = do(X_1 = 1)$ yielding $x^{post} = (1, 2)$. For the observation $x^{pre} = (0, 2)$ we can infer that U_Y must have been 1, since two increments are needed to get from 0 to 2. However, from the post-intervention observation $x^{post} = (1, 2)$ we cannot infer where the increment happened (U_Y or U_2). As a consequence, an optimal predictive model that only has access to x^{post} would predict that y^{post} for $x^{post} = (1, 2)$ could be 1 or 2 with equal likelihood. In contrast, with access to x^{pre} and the SCM we can infer that $y^{post} = 2$ since $U_Y = 1$.*

In the above example, given knowledge of the SCM, the pre-intervention observation x^{pre} and the performed action a we can already abduct U_Y perfectly and therefore correctly determine the post-intervention state of Y (even without access to the post-intervention observation x^{post}). In contrast, with the post-recourse observation alone it is impossible to reconstruct U_Y and therefore impossible to determine the post-intervention state of Y.[26] In the context of ICR this means that the observational predictor's post-recourse predictions are not directly linked with γ: they may not honor the implementation of actions with $\gamma^{ind} = 1$. As a consequence, we suggested to use the SCM for post-recourse prediction in Section 6.

[26] The optimal pre-recourse predictor $\hat{h}^*(x^{post})$ predicts 0.5 for both $y = 1$ and $y = 2$.

Supplements for Paper II: Scientific Inference With Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena

Appendix A Background on Models and Phenomena

We follow Bailer-Jones ([2003b], p61) and others (Achinstein 1974, Levy 2012, Contessa 2007) in seeing models as “an interpretative description of a phenomenon that facilitates perceptual as well as intellectual access to that phenomenon”, where a phenomenon describes a fact or event in nature that is subject to be researched (Bailer-Jones 2003a). Phenomenon and scientific models have been described as a continuous hierarchy with data living close to the phenomenon and the model close to theory (Suppes 1966). Models represent only some phenomenon aspects but not others (Ritchey 2012, Bailer-Jones 2003b, Frigg and Nguyen 2021); a good model is true to the aspects that are relevant to the model user (Bailer-Jones 2003b, Stachowiak 1973).

Appendix B Dataset

Figure 8 gives a descriptions of the different features and is copied from Cortez and Silva (2008). In our trained models, we only used the final G3 student grades. The data was collected during 2005 and 2006 from two public schools, from the Alentejo region in Portugal. The database is collected from a variety of sources from both school reports and questionnaires. Cortez and Silva (2008) integrated the information into a mathematics dataset (with 395 examples) and a Portuguese language dataset (649 records).

Attribute	**Description (Domain)**
sex	student's sex (binary: female or male)
age	student's age (numeric: from 15 to 22)
school	student's school (binary: *Gabriel Pereira* or *Mousinho da Silveira*)
address	student's home address type (binary: urban or rural)
Pstatus	parent's cohabitation status (binary: living together or apart)
Medu	mother's education (numeric: from 0 to 4^a)
Mjob	mother's job (nominal^b)
Fedu	father's education (numeric: from 0 to 4^a)
Fjob	father's job (nominal^b)
guardian	student's guardian (nominal: mother, father or other)
famsize	family size (binary: ≤ 3 or > 3)
famrel	quality of family relationships (numeric: from 1 – very bad to 5 – excellent)
reason	reason to choose this school (nominal: close to home, school reputation, course preference or other)
traveltime	home to school travel time (numeric: 1 – $<$ 15 min., 2 – 15 to 30 min., 3 – 30 min. to 1 hour or 4 – $>$ 1 hour).
studytime	weekly study time (numeric: 1 – $<$ 2 hours, 2 – 2 to 5 hours, 3 – 5 to 10 hours or 4 – $>$ 10 hours)
failures	number of past class failures (numeric: n if $1 \leq n < 3$, else 4)
schoolsup	extra educational school support (binary: yes or no)
famsup	family educational support (binary: yes or no)
activities	extra-curricular activities (binary: yes or no)
paidclass	extra paid classes (binary: yes or no)
internet	Internet access at home (binary: yes or no)
nursery	attended nursery school (binary: yes or no)
higher	wants to take higher education (binary: yes or no)
romantic	with a romantic relationship (binary: yes or no)
freetime	free time after school (numeric: from 1 – very low to 5 – very high)
goout	going out with friends (numeric: from 1 – very low to 5 – very high)
Walc	weekend alcohol consumption (numeric: from 1 – very low to 5 – very high)
Dalc	workday alcohol consumption (numeric: from 1 – very low to 5 – very high)
health	current health status (numeric: from 1 – very bad to 5 – very good)
absences	number of school absences (numeric: from 0 to 93)
G1	first period grade (numeric: from 0 to 20)
G2	second period grade (numeric: from 0 to 20)
G3	final grade (numeric: from 0 to 20)

a 0 – none, 1 – primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education.
b teacher, health care related, civil services (e.g. administrative or police), at home or other.

Figure 8: **Attributes in the Cortez and Silva (2008) dataset**.

Appendix C Tower Rule for Expectations

For arbitrary random variables $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ holds that

$$\mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X}}[\boldsymbol{Y} \mid \boldsymbol{X}] = \mathbb{E}_{\boldsymbol{Z}|\boldsymbol{X}}\Big[\mathbb{E}_{\boldsymbol{Y}|\boldsymbol{X},\boldsymbol{Z}}[\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{Z}] \mid \boldsymbol{X}\Big].$$

This is also known as the rule of total expectation. Intuitively it says that it doesn't matter if we directly take the expectation of $\boldsymbol{Y}$ on $\boldsymbol{X}$ or if we first take the expectation of $\boldsymbol{Y}$ conditioned on a set of random variables $\boldsymbol{X}, \boldsymbol{Z}$ that includes $\boldsymbol{X}$ and then, "integrate $\boldsymbol{Z}$ out".

Supplements for Paper III: Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process

Online Appendix for “Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process”

Christoph Molnar$^{\star 1,4[0000-0003-2331-868X]}$, Timo Freiesleben$^{\star 2[0000-0003-1338-3293]}$, Gunnar König$^{\star 1,3[0000-0001-6141-4942]}$, Julia Herbinger^{1,7}, Tim Reisinger^{1}, Giuseppe Casalicchio$^{1,7[0000-0001-5324-5966]}$, Marvin N. Wright$^{4,5,6[0000-0002-8542-6291]}$, and Bernd Bischl$^{1,7[0000-0001-6002-6980]}$

[1] Department of Statistics, LMU Munich, Munich, Germany
[2] Cluster of Excellence Machine Learning, Tübingen, Germany
[3] University of Vienna, Vienna, Austria
[4] Leibniz Institute for Prevention Research and Epidemiology, Bremen, Germany
[5] University of Bremen, Bremen, Germany
[6] University of Copenhagen, Copenhagen, Denmark
[7] Munich Center for Machine Learning (MCML)

A Background on Samplers

This is the more formal definition of a sampler: Let V and W be two random variables, we define a sampler as a function ϕ that maps an input $v \in \mathcal{V}$ to a density function on a space $\mathcal{W}$ i.e. $\phi : \mathcal{V} \to \{\psi \mid \psi \text{ density on } \mathcal{W}\}$. The two most common samplers in the context of PD and PFI are the marginal and the conditional sampler: the marginal sampler ϕ_{marg} maps every input $v \in \mathcal{V}$ to the density of W i.e. for all $v \in \mathcal{V}: \ \phi_{marg}(v) = \psi_W$; the conditional sampler ϕ_{cond} maps every input $v \in \mathcal{V}$ with $\psi_V(v) > 0$ to the conditional density of W i.e. for all $v \in \mathcal{V}: \ \phi_{cond}(v) = \psi_{W|V=v} = \frac{\psi_{W,V=v}}{\psi_{V=v}}$. As such, samples from $\phi_{marg}(v)$ follow $P(W)$, and samples from $\phi_{cond}(v)$ follow $P(W \mid V = v)$.

Like all model-agnostic interpretation techniques, both PD and PFI are based on sampling data and evaluating the model on these data [10]. Dependent on how we sample, we obtain different versions of PD and PFI and their results must be interpreted in a different way [8,7,4,13,1]. The two most common theoretical samplers in PD and PFI research are the marginal and the conditional sampler. The choice of the sampler should depend on the modeler's objective and the structure of the data. Under certain conditions, the marginal sampler allows to estimate causal effects [15]. However, for correlated input features the marginal sampler may create unrealistic data outside the training distribution, which is problematic if the goal is to draw inference about the DGP; under such conditions, the conditional sampler may be a better choice [5]. Samplers, especially conditional samplers, are generally not readily available, but must be

[⋆] equal contribution

learned with techniques such as conditional subgroups [7] or conditional density estimators [3,2,11,12,14,6]. The learning process of the sampler may introduce another source of uncertainty that we do not consider in this work; we discuss this limitation in the discussion section of the main paper.

B Bias and Variance of PD

The expected squared difference between model-PD and DGP-PD can be decomposed into bias and variance.

Proof.

$$\begin{aligned}\mathbb{E}_F[(PD_{\hat{f}}(x) - PD_f(x))^2] &= \mathbb{E}_F[PD_{\hat{f}}(x)^2] + \mathbb{E}_F[PD_f(x)^2] \\ &- 2\mathbb{E}_F[PD_{\hat{f}}(x)PD_f(x)] \\ &= \mathbb{V}_F[PD_{\hat{f}}(x)] + \mathbb{E}_F[PD_{\hat{f}}(x)]^2 \\ &+ PD_f(x)^2 - 2\mathbb{E}_F[PD_{\hat{f}}(x)PD_f(x)] \\ &= \underbrace{(PD_f(x) - \mathbb{E}_F[PD_{\hat{f}}(x)])^2}_{\text{Bias}} + \underbrace{\mathbb{V}_F[PD_{\hat{f}}(x)]}_{\text{Variance}}\end{aligned}$$

C Bias and Variance of PFI

The expected squared difference between model-PFI and DGP-PFI can be decomposed into bias and variance.

Proof.

$$\begin{aligned}\mathbb{E}_F[(PFI_{\hat{f}} - PFI_f)^2] &= \mathbb{E}_F[PFI_{\hat{f}}^2] + \mathbb{E}_F[PFI_f^2] \\ &- 2\mathbb{E}_F[PFI_{\hat{f}}PFI_f] \\ &= \mathbb{V}_F[PFI_{\hat{f}}] + \mathbb{E}_F[PFI_{\hat{f}}]^2 \\ &+ PFI_f^2 - 2\mathbb{E}_F[PFI_{\hat{f}}PFI_f] \\ &= (PFI_f - \mathbb{E}_F[PFI_{\hat{f}}])^2 + \mathbb{V}_F[PFI_{\hat{f}}] \\ &= Bias_F^2[PFI_{\hat{f}}] + \mathbb{V}_F[PFI_{\hat{f}}]\end{aligned}$$

D Model-PD Unbiasedness Regarding Theoretical PD

Proof. By the law of large numbers, the Monte Carlo integration converges with $r \to \infty$ to the true integral. Assuming we have a fixed x, r identically distributed random draws $\tilde{X}_C^{(1,x)}, \dots, \tilde{X}_C^{(r,x)} \sim \phi(x)$ and a model $\hat{f}$, the estimate is:

$$\begin{aligned}\mathbb{E}_{\tilde{X}_C}[\widehat{PD}_{\hat{f}}(x)] &= \mathbb{E}_{\tilde{X}_C^{(1,x)},\dots,\tilde{X}_C^{(r,x)}}\left[\frac{1}{r}\sum_{i=1}^{r}\hat{f}(x,\tilde{X}_C^{(i,x)})\right] \\ &= \frac{1}{r} r \mathbb{E}_{\tilde{X}_C}[\hat{f}(x,\tilde{X}_C)] \\ &= PD_{\hat{f}}(x)\end{aligned}$$

and therefore unbiased for the interval, i.e. the theoretical PD of the model.

E Model-PD Unbiasedness Regarding DGP-PD

Proof. Unbiasedness of the model $\hat{f}$ implies unbiasedness of the model-PD.

$$\begin{aligned}\mathbb{E}_F[PD_{\hat{f}}(x)] &\overset{Def}{=} \int_F \int_{\tilde{x}_c \in \tilde{\mathcal{X}}_C} \phi(x)(\tilde{x}_c)\hat{f}(x,\tilde{x}_c)\; d\tilde{x}_c\; dP(F) \\ &\overset{Fub}{=} \int_{\tilde{x}_c \in \tilde{\mathcal{X}}_C} \int_F \phi(x)(\tilde{x}_c)\hat{f}(x,\tilde{x}_c)\; dP(F)\; d\tilde{x}_c \\ &\overset{const.}{=} \int_{\tilde{x}_c \in \tilde{\mathcal{X}}_C} \phi(x)(\tilde{x}_c) \int_F \hat{f}(x,\tilde{x}_c)\; dP(F)\; d\tilde{x}_c \\ &\overset{unbiased}{=} \int_{\tilde{x}_c \in \tilde{\mathcal{X}}_C} \phi(x)(\tilde{x}_c) f(x,\tilde{x}_c)\; d\tilde{x}_c \\ &\overset{def}{=} PD_f(x)\end{aligned}$$

Fubini's theorem requires that $\int_{F,\tilde{X}_C} \mid \phi(x)(\tilde{X}_c)\hat{f}(\tilde{X}_c) \mid d\mathbb{P}_{F,X_C} < \infty$. One sufficient condition for this is when the model predictions have an upper bound $c :\mid \hat{f}(x) \mid < c < \infty$.

F Model-PFI Regarding theoretical PFI

Proof. As a function of random variables, the loss L itself is a random variable. We assume that the loss $L^{(i)}$ of observation i is a sample from the distribution of losses: $L^{(i)} \sim L$ and, similarly for the loss: $\tilde{L}^{(k,i)} \sim \tilde{L}$, where $L^{(i)} = L(y^{(i)}, \hat{f}(x^{(i)}))$ and $\tilde{L}^{(k,i)} = L(y^{(i)}, \hat{f}(\tilde{x}_S^{(k,i)}, x_C^{(i)}))$.

The expectation of our estimator is:

$$\begin{aligned}\mathbb{E}_{\tilde{X}_S X_S X_C Y}[\widehat{PFI}_{\hat{f}}] &= \mathbb{E}_{\tilde{X}_S X_S X_C Y}\left[\frac{1}{n_2}\sum_{i=1}^{n_2}(\frac{1}{r}\sum_{k=1}^{r}(\tilde{L}^{(k,i)} - L^{(i)}))\right] \\ &= \frac{1}{n_2} n_2 \mathbb{E}_{\tilde{X}_S X_S X_C Y}[((\frac{1}{r} r\tilde{L}) - L)] \\ &= \mathbb{E}_{\tilde{X}_S X_C Y}[\tilde{L}] - \mathbb{E}_{X_S X_C Y}[L] \\ &= PFI_{\hat{f}}\end{aligned}$$

In expectation, we retrieve the theoretical PFI of the model.

G PFI Biases for L2

In this proof, we use the conditional sampler ϕ_{cond} for both, the DGP-PFI and the model-PFI. Moreover, we assume that L is the squared loss $L(y, \hat{f}) = (y - \hat{f}(x))^2$ and that $\mathbb{E}[Y \mid X]$ can be described by f with some additive, irreducible, error ϵ with $\mathbb{E}(\epsilon) = 0$ and $\mathbb{V}(\epsilon) = \sigma^2$. To further examine the bias for the PFI, we apply the Bias-Variance Decomposition additionally on the loss itself: In addition, we use that $\mathbb{E}_{XY}[Y] = \mathbb{E}_X[f(X)]$, $\mathbb{V}_Y[Y] = \sigma^2$ and $\mathbb{E}[A^2] = \mathbb{V}[A] + \mathbb{E}[A]^2$. We first derive the bias-variance decomposition of (i) permuted loss and (ii) original loss and therefrom derive the expected PFI.

For the permuted loss (i):

$$\begin{aligned}
\mathbb{E}_{F\tilde{X}_SXY}[\tilde{L}] &= \mathbb{E}_{F\tilde{X}_SXY}[(Y - \tilde{\hat{f}})^2] \\
&= \mathbb{E}_{\tilde{X}_SXY}[Y^2 - 2Y\mathbb{E}_F[\tilde{\hat{f}}] + \mathbb{E}_F[\tilde{\hat{f}}^2]] \\
&= \mathbb{E}_{\tilde{X}_SXY}[Y^2 - 2Y\mathbb{E}_F[\tilde{\hat{f}}] + \mathbb{E}_F[\tilde{\hat{f}}]^2 + \mathbb{V}_F[\tilde{\hat{f}}]] \\
&= \mathbb{V}_Y[Y] + \mathbb{E}_{\tilde{X}_SX}[f^2 - 2f\mathbb{E}_F[\tilde{\hat{f}}] + \mathbb{E}_F[\tilde{\tilde{\hat{f}}}]^2 + \mathbb{V}_F[\tilde{\hat{f}}]] \\
&= \underbrace{\sigma^2}_{\text{Data Var}} + \mathbb{E}_{\tilde{X}_SX}\underbrace{\left[(f - \mathbb{E}_F[\tilde{\hat{f}}])^2\right]}_{\text{Bias}^2} + \mathbb{E}_{\tilde{X}_SX}\underbrace{[\mathbb{V}_F[\tilde{\hat{f}}]]}_{\text{Variance}}
\end{aligned}$$

For the original loss (ii):

$$\begin{aligned}
\mathbb{E}_{FXY}[L] &= \mathbb{E}_{FXY}[(Y - \hat{f})^2] \\
&= \mathbb{E}_{XY}[Y^2 - 2Y\mathbb{E}_F[\hat{f}] + \mathbb{E}_F[\hat{f}^2]] \\
&= \mathbb{E}_{XY}[Y^2 - 2Y\mathbb{E}_F[\hat{f}] + \mathbb{E}_F[\hat{f}]^2 + \mathbb{V}_F[\hat{f}]] \\
&= \mathbb{V}_Y[Y] + \mathbb{E}_X[f^2 - 2f\mathbb{E}_F[\hat{f}] + \mathbb{E}_F[\hat{f}]^2 + \mathbb{V}_F[\hat{f}]] \\
&= \underbrace{\sigma^2}_{\text{Data Var}} + \mathbb{E}_X\underbrace{\left[(f - \mathbb{E}_F[\hat{f}])^2\right]}_{\text{Bias}^2} + \mathbb{E}_X\underbrace{[\mathbb{V}_F(\hat{f})]}_{\text{Variance}}
\end{aligned}$$

The expected PFI for feature X_S then is:

$$\begin{aligned}
\mathbb{E}_F[PFI_{\hat{f}}] &= \mathbb{E}_{F\tilde{X}_SXY}[\tilde{L}] - \mathbb{E}_{FXY}[L] \\
&\overset{(i)+(ii)}{=} \sigma^2 + \mathbb{E}_{\tilde{X}_SX}\left[(f - \mathbb{E}_F[\tilde{\hat{f}}])^2\right] + \mathbb{E}_{\tilde{X}_SX}[\mathbb{V}_F(\tilde{\hat{f}})] \\
&- (\sigma^2 + \mathbb{E}_X\left[(f - \mathbb{E}_F[\hat{f}])^2\right] + \mathbb{E}_X[\mathbb{V}_F(\hat{f})]) \\
&= \mathbb{E}_{\tilde{X}_SX}\left[(f - \mathbb{E}_F[\tilde{\hat{f}}])^2\right] - \mathbb{E}_X\left[(f - \mathbb{E}_F[\hat{f}])^2\right] \\
&+ \mathbb{E}_{\tilde{X}_SX}[\mathbb{V}_F[\tilde{\hat{f}}]] - \mathbb{E}_X[\mathbb{V}_F[\hat{f}]]
\end{aligned}$$

We can derive the same L2 decomposition for the DGP-PFI by replacing $\hat{f}$ with f in the equation above. This yields $PFI_f = \mathbb{E}_{\tilde{X}_S X}[(f(X) - f(\tilde{X}_S, X_C))^2]$, since $\mathbb{V}_F[f] = \mathbb{V}_F[\tilde{f}] = 0$ and $\mathbb{E}_F[f] = f$ and $\mathbb{E}_F[\tilde{f}] = \tilde{f}$.

The bias of the model-PFI compared to the DGP-PFI is:

$$
\begin{aligned}
\mathbb{E}_F[PFI_{\hat{f}}] - PFI_f &= \underbrace{\mathbb{E}_{\tilde{X}_S X}[(f - \mathbb{E}_F[\tilde{\hat{f}}])^2 - (f - \tilde{f})^2]}_{\text{Permutation Loss Bias}} && (1)\\
&\quad - \underbrace{\mathbb{E}_X\left[(f - \mathbb{E}_F[\hat{f}])^2\right]}_{\text{(Learner Bias)}^2} + \underbrace{\mathbb{E}_{\tilde{X}_S X}[\mathbb{V}_F[\hat{f}]] - \mathbb{E}_X[\mathbb{V}_F[\hat{f}]]}_{\text{Variance Inflation}} && (2)\\
&\overset{unbiased}{=} \underbrace{\mathbb{E}_{\tilde{X}_S X}[\mathbb{V}_F[\hat{f}]] - \mathbb{E}_X[\mathbb{V}_F[\hat{f}]]}_{\text{Variance Inflation}} && (3)\\
&\overset{\tilde{X}_S \sim X_S | X_C}{=} 0 && (4)
\end{aligned}
$$

The permutation loss bias and the squared learner bias are zero due to the unbiasedness assumption, i.e. $\mathbb{E}_F[\hat{f}] = f$. The variance inflation term is zero if $\tilde{X}_S \sim X_S \mid X_C$, which is here the case due to conditional sampling.

H conditional DGP-PFI minus model-PFI for L2

In this proof, we use the conditional sampler ϕ_{cond} for both, the DGP-PFI and the model-PFI.

$$
\begin{aligned}
PFI_f - PFI_{\hat{f}} &= \mathbb{E}_{\tilde{X_S} X_C Y}[(Y - f)^2] - \mathbb{E}_{X_S X_C Y}[(Y - f)^2]\\
&\quad - \Big(\mathbb{E}_{\tilde{X_S} X_C Y}[(Y - \hat{f})^2] - \mathbb{E}_{X_S X_C Y}[(Y - \hat{f})^2]\Big)\\
&= \underbrace{\Big(\mathbb{E}_{X_S X_C Y}[(Y - \hat{f})^2] - \mathbb{E}_{X_S X_C Y}[(Y - f)^2]\Big)}_{\text{T1:=}}\\
&\quad + \underbrace{\Big(\mathbb{E}_{\tilde{X_S} X_C Y}[(Y - f))^2] - \mathbb{E}_{\tilde{X_S} X_C Y}[(Y - \hat{f})^2]\Big)}_{\text{T2:=}}
\end{aligned}
$$

We know that for any $g : X \to Y$ holds:

$$\mathbb{E}_{X,Y}[(Y - g)^2] = \mathbb{E}_X[\mathbb{V}_{Y|X}[Y]] + \mathbb{E}_X[(\mathbb{E}_{Y|X}[Y] - g)^2]$$

Since $f = \mathbb{E}_{Y|X_S,X_C}[Y]$ we can conclude for our first term T1 that:

$$
\begin{aligned}
\text{T1} &= \mathbb{E}_{X_S X_C}[\mathbb{V}_{Y|X_S,X_C}[Y]] + \mathbb{E}_{X_S X_C}[(f - \hat{f})^2]\\
&\quad - \Big(\mathbb{E}_{X_S X_C}[\mathbb{V}_{Y|X_S,X_C}[Y]] + \underbrace{\mathbb{E}_{X_S X_C}[(f - f)^2]}_{=0}\Big)\\
&= \mathbb{E}_{X_S X_C}[(f - \hat{f})^2]
\end{aligned}
$$

We apply the same strategy to T2. Moreover, $Y \perp\!\!\!\perp \tilde{X}_S \mid X_C$.

$$\begin{aligned}
\text{T2} &= \mathbb{E}_{\tilde{X_S} X_C}[\mathbb{V}_{Y|\tilde{X}_S,X_C}[Y]] + \mathbb{E}_{\tilde{X_S} X_C}[(\mathbb{E}_{Y|\tilde{X}_S,X_C}[Y] - f)^2] \\
&\quad - \Big(\mathbb{E}_{\tilde{X_S} X_C}[\mathbb{V}_{Y|\tilde{X}_S,X_C}[Y]] + \mathbb{E}_{\tilde{X_S} X_C}[(\mathbb{E}_{Y|\tilde{X}_S,X_C}[Y] - \hat{f})^2]\Big) \\
&= \mathbb{E}_{\tilde{X_S} X_C}[(\mathbb{E}_{Y|X_C}[Y] - f)^2] - \mathbb{E}_{\tilde{X_S} X_C}[(\mathbb{E}_{Y|X_C}[Y] - \hat{f})^2]
\end{aligned}$$

If we now set together the two terms again and use in the first step that $P(X_S, X_C) = P(\tilde{X}_S, X_C)$, we obtain:

$$\begin{aligned}
\text{T1+T2} &= \mathbb{E}_{X_S X_C}[(f - \hat{f})^2] + \mathbb{E}_{X_S X_C}[(\mathbb{E}_{Y|X_C}[Y] - f)^2] \\
&\quad - \mathbb{E}_{X_S X_C}[(\mathbb{E}_{Y|X_C}[Y] - \hat{f})^2] \\
&= \mathbb{E}_{X_S X_C}\Big[f^2 - 2f\hat{f} + \hat{f}^2 + \mathbb{E}_{Y|X_C}[Y]^2 - 2\mathbb{E}_{Y|X_C}[Y]f + f^2 \\
&\quad - \mathbb{E}_{Y|X_C}[Y]^2 + 2\mathbb{E}_{Y|X_C}[Y]\hat{f} - \hat{f}^2\Big] \\
&= 2\mathbb{E}_{X_S X_C}\Big[(f^2 - \mathbb{E}_{Y|X_C}[Y]f) - (f\hat{f} - \mathbb{E}_{Y|X_C}[Y]\hat{f})\Big] \\
&= 2\mathbb{E}_{X_C}\Big[\mathbb{E}_{X_S|X_C}[(f^2 - \mathbb{E}_{Y|X_C}[Y]f) - (f\hat{f} - \mathbb{E}_{Y|X_C}[Y]\hat{f})]\Big] \\
&\overset{*}{=} 2\mathbb{E}_{X_C}\Big[(\mathbb{E}_{X_S|X_C}[f^2] - \mathbb{E}_{Y|X_C}[Y]\mathbb{E}_{X_S|X_C}[f]) \\
&\quad - (\mathbb{E}_{X_S|X_C}[f\hat{f}] - \mathbb{E}_{Y|X_C}[Y]\mathbb{E}_{X_S|X_C}[\hat{f}])\Big] \\
&\overset{**}{=} 2\mathbb{E}_{X_C}\Big[(\mathbb{E}_{X_S|X_C}[f^2] - \mathbb{E}_{X_S|X_C}[f]^2) \\
&\quad - (\mathbb{E}_{X_S|X_C}[f\hat{f}] - \mathbb{E}_{X_S|X_C}[\hat{f}]\mathbb{E}_{X_S|X_C}[f])\Big] \\
&= 2\mathbb{E}_{X_C}\big[\mathbb{V}_{X_S|X_C}[f] - Cov_{X_S|X_C}[f, \hat{f}]\big]
\end{aligned}$$

At *, we use the fact that the random variable $\mathbb{E}_{Y|X_C}[Y]$ is measurable by the σ-Algebra generated from X_C, and we are inclined to pull it out of the expectation. In **, we use that from $f = \mathbb{E}_{Y|X_S,X_C}[Y]$ follows $\mathbb{E}_{X_S|X_C}[f] = \mathbb{E}_{Y|X_C}[Y]$.

I CI simulation results

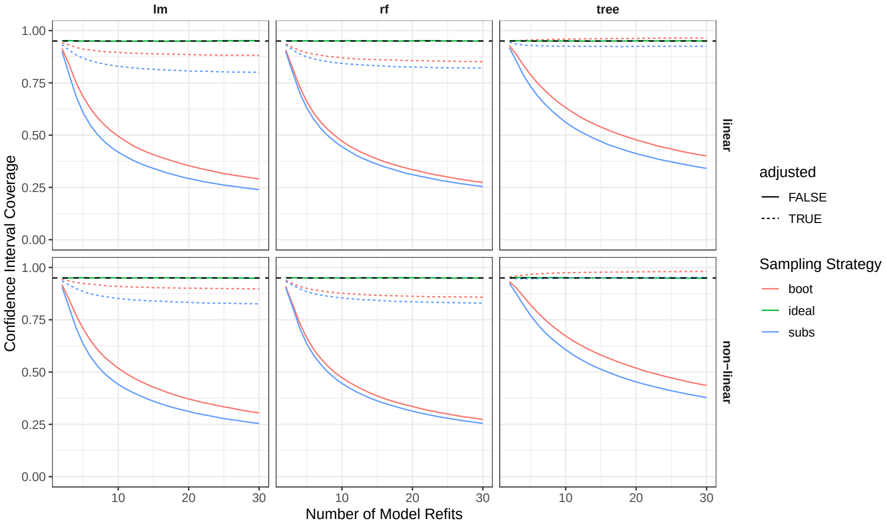


Figure I.1: CI coverage for PD with n=100.

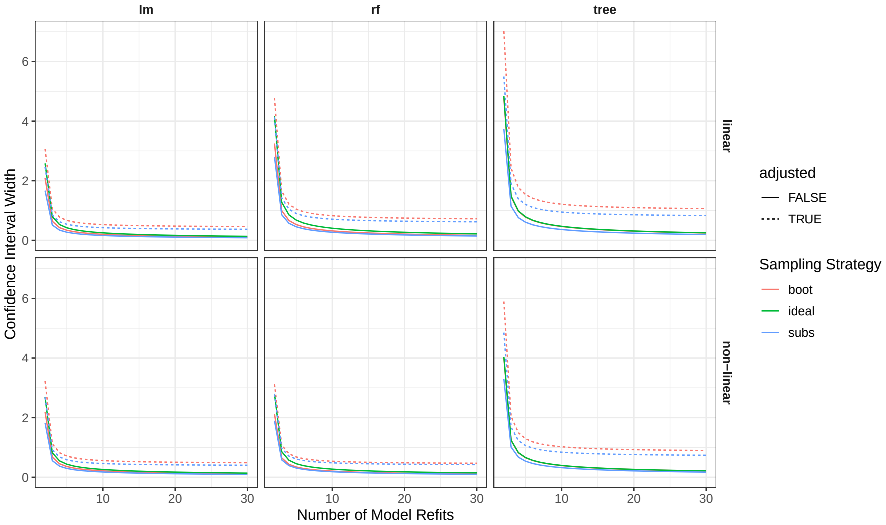


Figure I.2: CI width for PD with n=100.

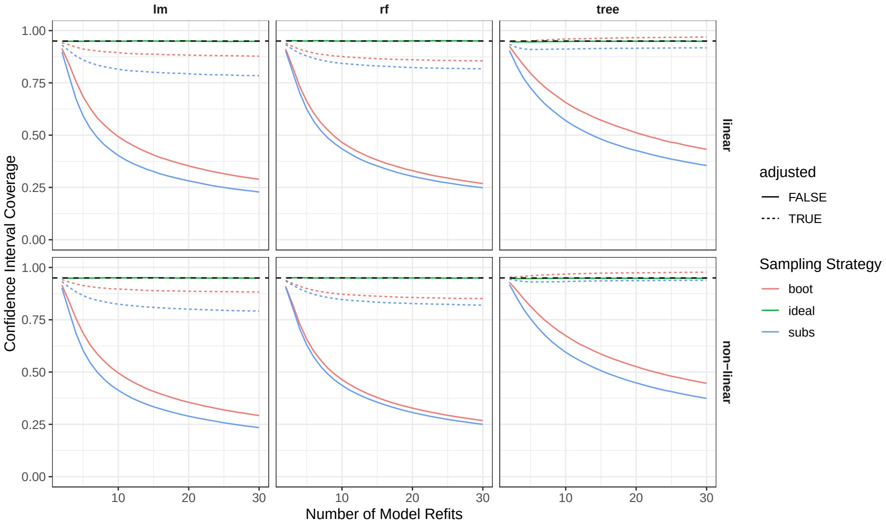


Figure I.3: CI coverage for PD with n=1,000.

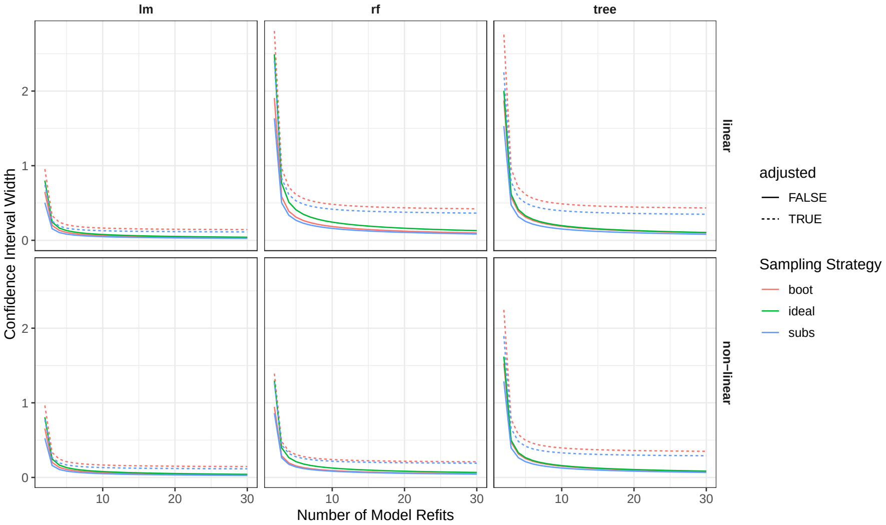


Figure I.4: CI width for PD with n=1,000.

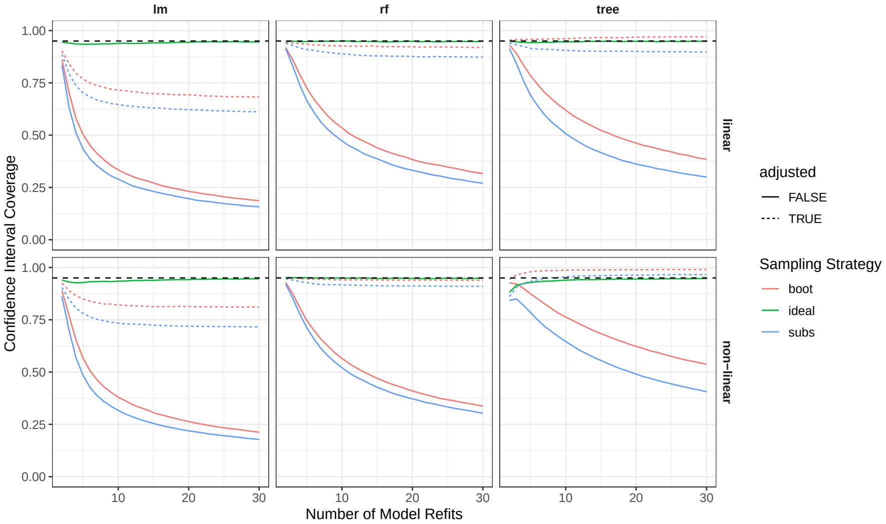


Figure I.5: CI coverage for PFI with n=100.

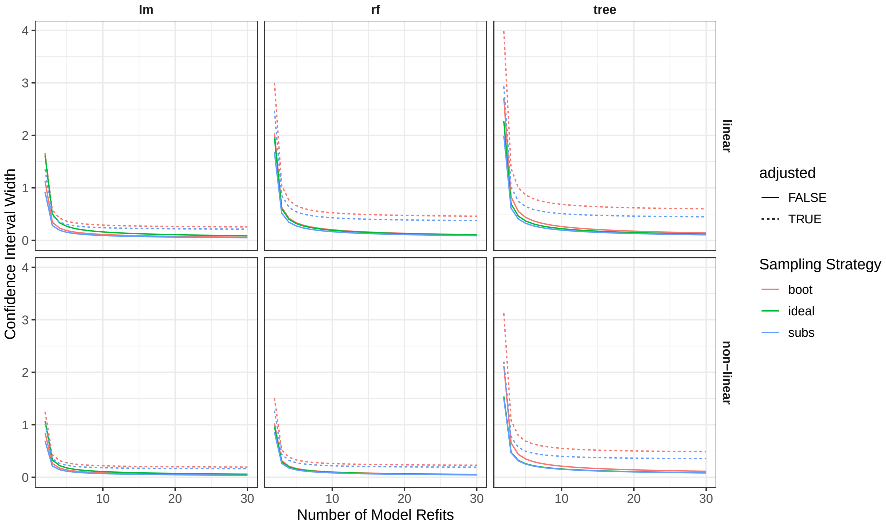


Figure I.6: CI width for PFI with n=100.

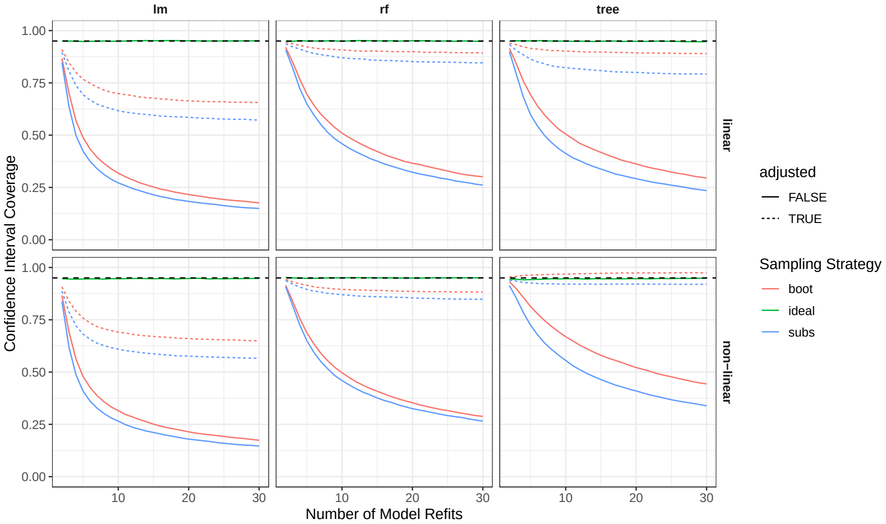


Figure I.7: CI coverage for PFI with n=1,000.

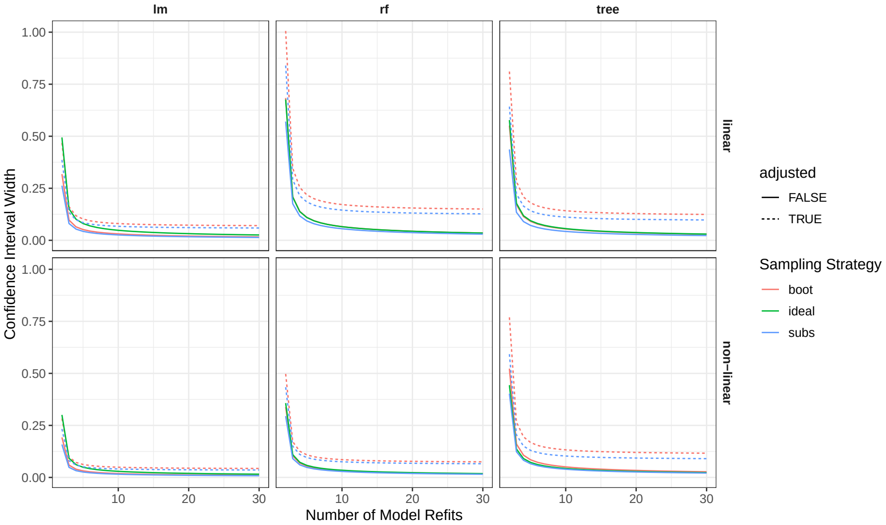


Figure I.8: CI width for PFI with n=1,000.

J Theoretical background of model-based uncertainty

[9] leverage the kernel of GPs to analytically calculate the model-based uncertainty contained in the PD function. Let $\hat{f}$ be a GP and $\hat{\boldsymbol{m}}(x) = \left(\hat{m}(x, x_C^{(i)})\right)_{i=1,\ldots,n_2}$ its estimated posterior mean and $\hat{\boldsymbol{K}}(x) = \left(\hat{k}\big((x, x_C^{(i)}), (x, x_C^{(j)})\big)\right)_{i,j=1,\ldots,n_2}$ its estimated posterior covariance on the test set D_{n_2} for fixed feature values $x \in X_S$. The PD estimate $\widehat{PD}$ of $\hat{f}$ can be seen as a random variable. Thus, the PD for the posterior mean function is given by the expected value of $\widehat{PD}$:

$$\mathbb{E}_{\hat{f}}\left[\widehat{PD}(x)\right] = \mathbb{E}_{\hat{f}}\left[\frac{1}{n_2}\sum_{i=1}^{n_2}\hat{f}(x, x_C^{(i)})\right] = \frac{1}{n_2}\sum_{i=1}^{n_2}\hat{m}(x, x_C^{(i)}). \tag{5}$$

The variance of the PD is estimated accordingly and can be calculated straightforwardly by leveraging the posterior covariance of the GP:

$$\mathbb{V}_{\hat{f}}\left[\widehat{PD}(x)\right] = \mathbb{V}_{\hat{f}}\left[\frac{1}{n_2}\sum_{i=1}^{n_2}\hat{f}(x, x_C^{(i)})\right] = \frac{1}{{n_2}^2}\mathbf{1}^\top\hat{\boldsymbol{K}}(x)\mathbf{1}. \tag{6}$$

Since the n_2 predictors $\hat{f}(x, x_C^{(i)})$ of the GP follow a Gaussian distribution, their sum is also normally distributed. Hence, we can construct confidence bands for the mean estimate in Eq. (5) by using the variance estimate in Eq. (6) together with the respective $1-\alpha/2$ quantiles of the Gaussian distribution. This approach is applicable to any models (including non-GPs) that provide a fully specified covariance matrix between the predictions.

As Eq. (6) solely quantifies the variance w.r.t. the model given the observed data, the resulting confidence bands only capture model variance but not the variance induced by MC integration.

References

1. Apley, D.W., Zhu, J.: Visualizing the effects of predictor variables in black box supervised learning models. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **82**(4), 1059–1086 (2020). https://doi.org/10.1111/rssb.12377
2. Bashtannyk, D.M., Hyndman, R.J.: Bandwidth selection for kernel conditional density estimation. Computational Statistics & Data Analysis **36**(3), 279–298 (2001)
3. Bishop, C.M.: Mixture density networks. Tech. rep., Aston University (1994)
4. Fisher, A., Rudin, C., Dominici, F.: All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. Journal of Machine Learning Research **20**(177), 1–81 (2019)
5. Freiesleben, T., König, G., Molnar, C., Tejero-Cantero, A.: Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena. arXiv preprint arXiv:2206.05487 (2022)
6. Hothorn, T., Zeileis, A.: Predictive distribution modeling using transformation forests. Journal of Computational and Graphical Statistics **30**(4), 1181–1196 (2021)
7. Molnar, C., König, G., Bischl, B., Casalicchio, G.: Model-agnostic feature importance and effects with dependent features–a conditional subgroup approach. arXiv preprint arXiv:2006.04628 (2020)
8. Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C.A., Casalicchio, G., Grosse-Wentrup, M., Bischl, B.: General pitfalls of model-agnostic interpretation methods for machine learning models, pp. 39–68. Springer International Publishing, Cham (2022). https://doi.org/10.1007/978-3-031-04083-2_4, https://doi.org/10.1007/978-3-031-04083-2_4
9. Moosbauer, J., Herbinger, J., Casalicchio, G., Lindauer, M., Bischl, B.: Explaining hyperparameter optimization via partial dependence plots. Advances in Neural Information Processing Systems **34**, 2280–2291 (2021)
10. Scholbeck, C.A., Molnar, C., Heumann, C., Bischl, B., Casalicchio, G.: Sampling, intervention, prediction, aggregation: A generalized framework for model-agnostic interpretations. Communications in Computer and Information Science p. 205–216 (2020). https://doi.org/10.1007/978-3-030-43823-4_18
11. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. Advances in neural information processing systems **28** (2015)
12. Trippe, B.L., Turner, R.E.: Conditional density estimation with bayesian normalising flows. arXiv preprint arXiv:1802.04908 (2018)
13. Watson, D.S., Wright, M.N.: Testing conditional independence in supervised learning algorithms. Machine Learning **110**, 2107–2129 (2021). https://doi.org/10.1007/s10994-021-06030-6
14. Winkler, C., Worrall, D., Hoogeboom, E., Welling, M.: Learning likelihoods with conditional normalizing flows. arXiv preprint arXiv:1912.00042 (2019)
15. Zhao, Q., Hastie, T.: Causal interpretations of black-box models. Journal of Business & Economic Statistics **39**(1), 272–281 (2021). https://doi.org/10.1080/07350015.2019.1624293

Supplements for Paper VI: Efficient SAGE Estimation via Causal Structure Learning

A PROOF OF THEOREM 1

Theorem 1. *For ℓ being cross-entropy loss or the mean-squared error, f^* the respective optimal predictor and ν_{ℓ,f^*} the corresponding SAGE value function, it holds that*

$$X_j \perp Y|\boldsymbol{X}_S \Rightarrow \nu_{\ell,f^*}(\boldsymbol{X}_{S\cup j}) - \nu_{\ell,f^*}(\boldsymbol{X}_S) = 0.$$

Proof. Mean Squared Error: Covert et al. (2020) show that for ℓ being the mean squared error and f^* the corresponding optimal predictor it holds that:

$$\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S) = \mathbb{E}[\text{Var}(Y|\mathbf{X}_S)] - \mathbb{E}[\text{Var}(Y|\mathbf{X}_{S\cup j})]$$

Under conditional independence $Y \perp X_j|\mathbf{X}_S$ it follows that

$$\begin{aligned}\mathbb{E}[\text{Var}(Y|\mathbf{X}_{S\cup j})] &= \mathbb{E}[\mathbb{E}[\text{Var}(Y|\mathbf{X}_{S\cup j})|\mathbf{X}_S]] \\ &= \mathbb{E}[\text{Var}(Y|\mathbf{X}_S)]\end{aligned}$$

and consequently $Y \perp X_j|\mathbf{X}_S \Rightarrow \nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S) = 0$.

Cross Entropy: Covert et al. (2020) show that given cross entropy as loss and the corresponding loss optimal predictor f^* it holds that:

$$\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S) = I(Y; X_j|\mathbf{X}_S)$$

Mutual information $I(Y; X_j|\mathbf{X}_S)$ is zero if and only if $Y \perp X_j|\mathbf{X}_S$. Consequently $\nu(\mathbf{X}_{S\cup j}) - \nu(\mathbf{X}_S) = 0 \Leftrightarrow X_j \perp Y|\mathbf{X}_S$. □

B SAGE VALUE PROPERTIES

As mentioned in Section 3, SAGE values satisfy certain fairness properties that are deduced from those valid for Shapley values (Covert et al., 2020). While not explicitly named after the Shapley value properties (*efficiency*, the *dummy property*, *symmetry*, *monotonicity*, *linearity*) we employ these terms for the SAGE properties for simplicity:

1. *Efficiency*: $\sum_{j=1}^{d} \phi_j(\nu) = \nu(\mathbf{X})$, where $\mathbf{X}$ is the set of all features.
2. *Dummy property*: $\phi_j(\nu) = 0$ if $X_j \perp \hat{f}(\mathbf{X})|\mathbf{X}_S$ for all $S \subseteq \{1, ..., d\} \setminus j$.
3. *Symmetry*: $\nu(\mathbf{X}_{S\cup j}) = \nu(\mathbf{X}_{S\cup i})$ for two variables X_j and X_i with a deterministic relationship.
4. *Monotonicity*: For two target variables Y, Y' and corresponding models $\hat{f}$, $\hat{f}'$: $\phi_j(\nu_{\hat{f}}) \geq \phi_j(\nu_{\hat{f}'})$ if $\nu_{\hat{f}}(\mathbf{X}_{S\cup j}) - \nu_{\hat{f}}(\mathbf{X}_S) \geq \nu_{\hat{f}'}(\mathbf{X}_{S\cup j}) - \nu_{\hat{f}'}(\mathbf{X}_S)$ for all $S \subseteq \{1, ..., d\} \setminus j$.
5. From *Linearity*: $\phi_j(\nu) = \mathbb{E}_{\mathbf{X},Y}[\phi_j(\nu_{\hat{f},x,y})]$, where $\phi_j(\nu_{\hat{f},x,y})$ is the Shapley value of the game $\nu_{\hat{f},x,y}(\mathbf{X}_S) = \ell(\hat{f}_\emptyset(\mathbf{X}_\emptyset), y) - \ell(\hat{f}_S(\mathbf{X}_S), y)$
6. SAGE values are invariant to invertible mappings applied to the input, e.g. they are the same for original input data and and their log values.

C GRAPH BENCHMARK

In this section, we provide detailed information about the graphs employed in Section 5, the graph learning algorithms and the graph benchmark. Additionally, we present results derived from the HC algorithm for CSL.

C.1 Overview of Graphs

In Table 1 we provide an overview of all twelve graphs used in Section 5, the randomly sampled target, the adjacency degree of the target and the share of d-separations w.r.t. the target. This gives further insight into the relation of graph sparsity, degree of target and share of d-separations. The latter can be regarded as the potential relative runtime decrease for SAGE approximation.

Table 1: Overview of all twelve graphs used in Section 5, the randomly sampled target, the adjacency degree of the target and the share of d-separations w.r.t. the target.

GRAPH (AVG. DEGREE)	TARGET	DEGREE OF TARGET	SHARE OF $\perp_{\mathcal{G}}$
DAG$_s$(2)	8	2	0.556
DAG$_s$(3)	1	2	0.357
DAG$_s$(4)	1	4	0.283
DAG$_{sm}$(2)	17	1	0.765
DAG$_{sm}$(3)	2	1	0.623
DAG$_{sm}$(4)	16	4	0.185
DAG$_m$(2)	4	1	0.961
DAG$_m$(3)	32	5	0.556
DAG$_m$(4)	2	3	0.274
DAG$_l$(2)	4	3	0.632
DAG$_l$(3)	66	3	0.552
DAG$_l$(4)	66	7	0.151

Table 2 shows the hyperparameter settings used for CSL relying on the *bnlearn* package (Scutari, 2010) for R (R Core Team, 2022).

Table 2: Hyperparameters Used for Graph Learning

ALGORITHM	HYPERPARAMETERS
HC	Max. iterations ∞, max. in-degree: ∞; score: BIC
TABU	Size of list: 10; Max. iterations ∞, max. in-degree: ∞; score: BIC

C.2 MC Sampling for d-separation Inference

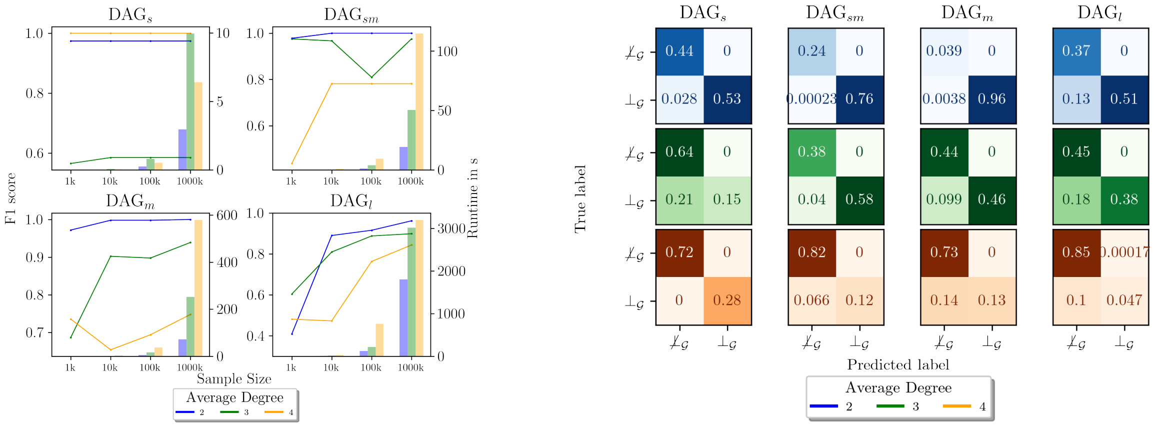
In Algorithm 2 we explicate how we inferred the number of true positive, false positive, true negative and false negative d-separations within an estimated graph and especially how we dealt with the exponential number of potential conditioning sets for the larger graphs.

Algorithm 2: Monte Carlo Sampling for d-separation Inference

```
Input: True graph $\mathcal{G}^*$ and estimated graph $\mathcal{G}$ over node set $\{X_1, X_2, ...X_d, Y\}$ with target node $Y$; Number of MC samples $n_{mc}$
Output: True positives, true negatives, false positives and false negatives for inferred $d$-separations in $\mathcal{G}$: TP, TN, FP, FN
Set $TP = TN = FP = FN = 0$
for $m = 1, ..., n_{mc}$ do
    Randomly draw a node $X_j$ from $\{X_1, X_2, ...X_d\}$
    Randomly draw size $n_s$ of conditioning set $\mathbf{X}_S$ from discrete probability distribution $P(n_s = i) = \frac{\binom{d-1}{i}}{2^{d-1}}, i \in \{0, ..., d-1\}$
    Randomly draw elements $X_i, i = 1, ...n_s$, from $\{X_1, X_2, ...X_d\} \setminus X_j$ without replacement and set $\mathbf{X}_S = \{X_i\}_{i=1,...,n_s}$
    if $X_j \perp_{\mathcal{G}^*} Y | \boldsymbol{X}_S$ then
        if $X_j \perp_{\mathcal{G}} Y | \boldsymbol{X}_S$ then
            TP = TP+1
        else
            FN = FN+1
        end
    else
        if $X_j \not\perp_{\mathcal{G}} Y | \boldsymbol{X}_S$ then
            TN = TN+1
        else
            FP = FP+1
        end
    end
end
Return: TP, TN, FP, FN
```

C.3 Results - HC

In Figure 7 we show the results of the graph learning benchmark for HC in contrast to those from Section 5. As HC never performed better but for some experiments worse than TABU, we chose the latter for the use in d-SAGE.



(a) F1 scores for d-separation (lines, left y-axes) and runtime of graph learning (bars, right y-axes) using HC depending on sample size.

(b) Confusion matrix for true and predicted d-connections ($\not\perp_{\mathcal{G}}$) and d-separations ($\perp_{\mathcal{G}}$) based on HC with $n = 10,000$ for all twelve graphs.

Figure 7: Results from graph learning benchmark for HC algorithm.

D SAGE - EXPERIMENTS

In this section, we briefly explain the experiment setup and afterwards present missing results. For our analysis, we fitted two models, LM and RF, for every dataset relying on the same targets that were sampled randomly for the analysis of d-separations in a graph. We relied on $n_{train} = 8000$ for model fitting and $n_{test} = 2000$ for model evaluation (the same $n = 10000$ data points as used for graph fitting and SAGE inference). We then used the data to estimate SAGE and d-SAGE five times, i.e. we were provided five approximations of (d-)SAGE for every graph and model, which were then used to provide error bounds. The $\Delta_{j|S}$ plots rely on skipped evaluations of each of these runs.

In Table 3 we provide performance measures of the models and in Appendix D.1 the plots pertaining to experiments not shown in Section 5 are displayed. Note that Table 3 highlights that RF performs slightly worse than the optimal LM throughout all settings and with regard to the MSE and R^2.

Table 3: Details of Linear Models (LMs) and Random Forests (RF); Random Forests based on 100 Tree Estimators.

DATA (AVERAGE DEGREE)	$\mathbf{n}_{train}$; $\mathbf{n}_{test}$	**MSE**$_{LM}$	$\mathbf{R}^2_{LM}$	**MSE**$_{RF}$	$\mathbf{R}^2_{RF}$
DAG$_s$(2)	8000; 2000	0.541	0.495	0.572	0.466
DAG$_{sm}$(2)	8000; 2000	0.035	0.963	0.038	0.960
DAG$_m$(2)	8000; 2000	0.474	0.522	0.498	0.498
DAG$_l$(2)	8000; 2000	0.070	0.930	0.103	0.897
DAG$_s$(3)	8000; 2000	0.382	0.616	0.480	0.517
DAG$_{sm}$(3)	8000; 2000	0.072	0.926	0.078	0.921
DAG$_m$(3)	8000; 2000	0.089	0.914	0.174	0.832
DAG$_l$(3)	8000; 2000	0.065	0.938	0.082	0.922
DAG$_s$(4)	8000; 2000	0.101	0.902	0.161	0.843
DAG$_{sm}$(4)	8000; 2000	0.075	0.925	0.086	0.914
DAG$_m$(4)	8000; 2000	0.163	0.840	0.194	0.810
DAG$_l$(4)	8000; 2000	0.004	0.996	0.059	0.943

D.1 Results - SAGE and d-SAGE

In this section we provide the same results as in Section 5 for all missing setups and both models, LM and RF as well as the top fifteen values for the setup presented in Section 5. Overall, we can confirm our findings in the different settings.

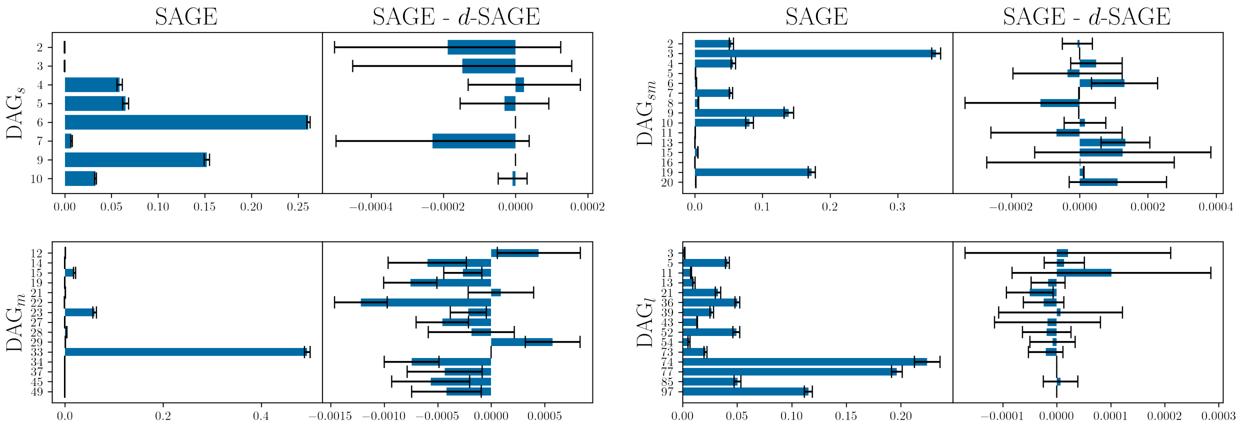
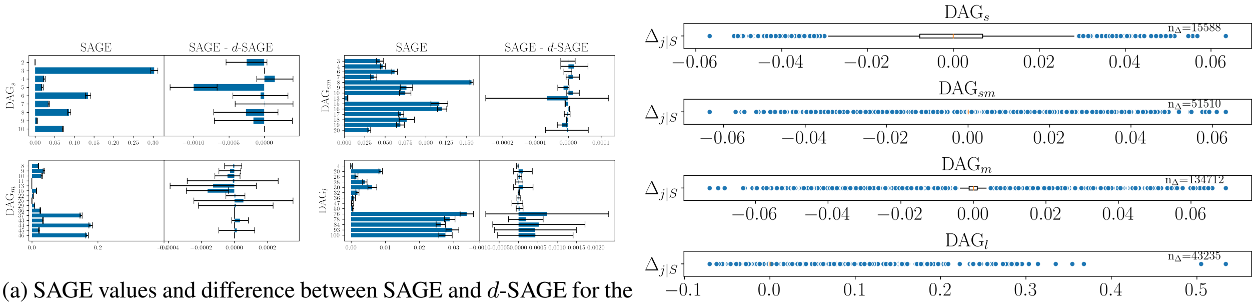


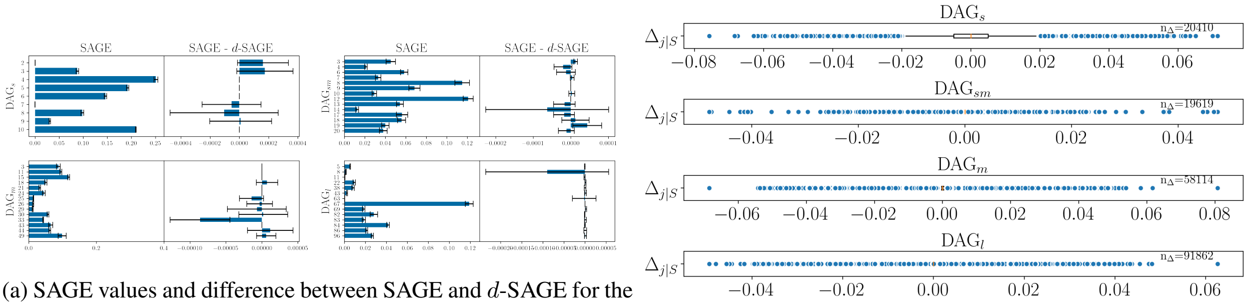
Figure 8: SAGE values and difference between SAGE and d-SAGE for the fifteen (all for DAG$_s$) largest values for optimal models for DAGs with average degree two.



(a) SAGE values and difference between SAGE and d-SAGE for the fifteen (all for DAG$_s$) largest values for optimal models.

(b) Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

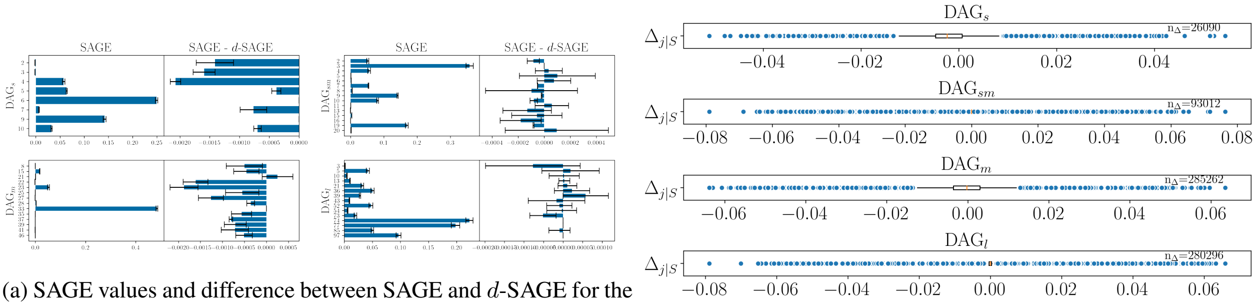
Figure 9: Results on the estimation quality for d-SAGE based on each DAG with average degree three and the LM.



(a) SAGE values and difference between SAGE and d-SAGE for the fifteen largest (all for DAG$_s$) values for optimal models

(b) Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

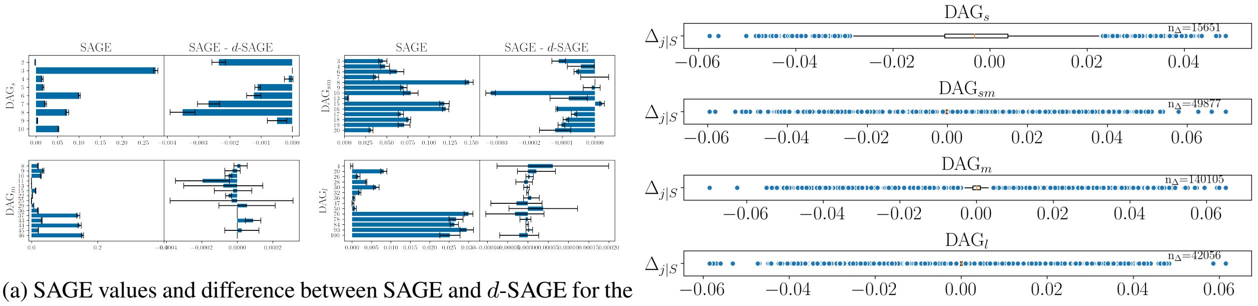
Figure 10: Results on the estimation quality for d-SAGE based on each DAG with average degree four and the LM.



(a) SAGE values and difference between SAGE and d-SAGE for the fifteen (all for DAG$_s$) largest values for optimal models.

(b) Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

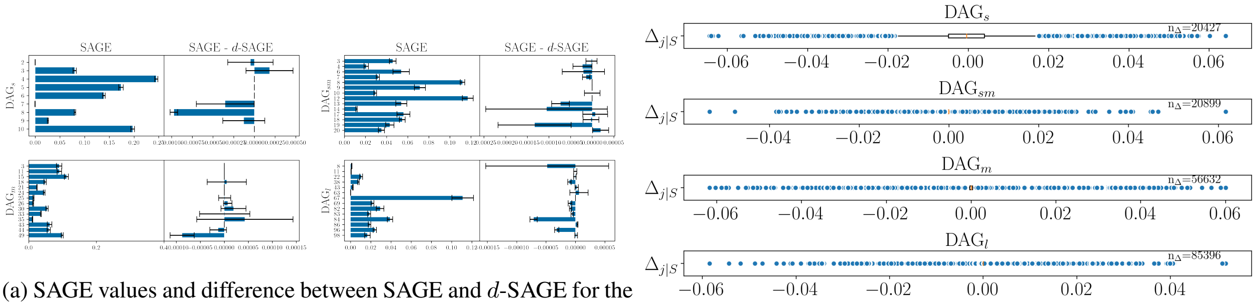
Figure 11: Results on the estimation quality for d-SAGE based on each DAGs with average degree two and the RF.



(a) SAGE values and difference between SAGE and d-SAGE for the fifteen (all for DAG$_s$) largest values for optimal models.

(b) Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

Figure 12: Results on the estimation quality for d-SAGE based on each DAG with average degree three and the RF.



(a) SAGE values and difference between SAGE and d-SAGE for the fifteen (all for DAG$_s$) largest values for optimal models.

(b) Boxplots showing the distribution of $\Delta_{j|S}$ for the skipped surplus evaluations.

Figure 13: Results on the estimation quality for d-SAGE based on each DAG with average degree four and the RF.

E CONVERGENCE PLOTS

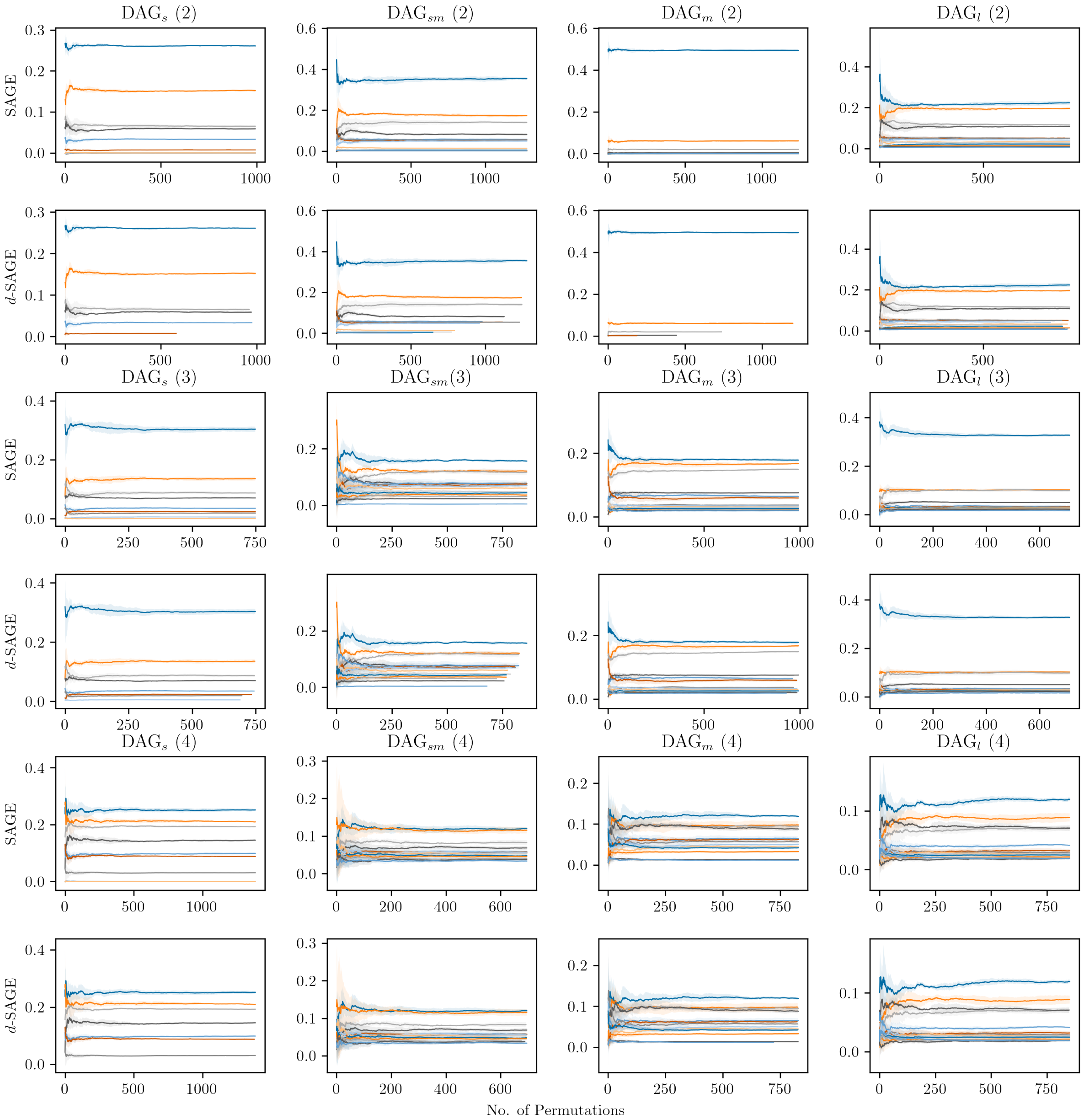


Figure 14: Convergence of largest fifteen SAGE and d-SAGE values for optimal models (LM) for every DAG (average adjacency degree). Each colour represents the same feature in SAGE and d-SAGE plots for a given graph (if present in both). Legend omitted for readability.

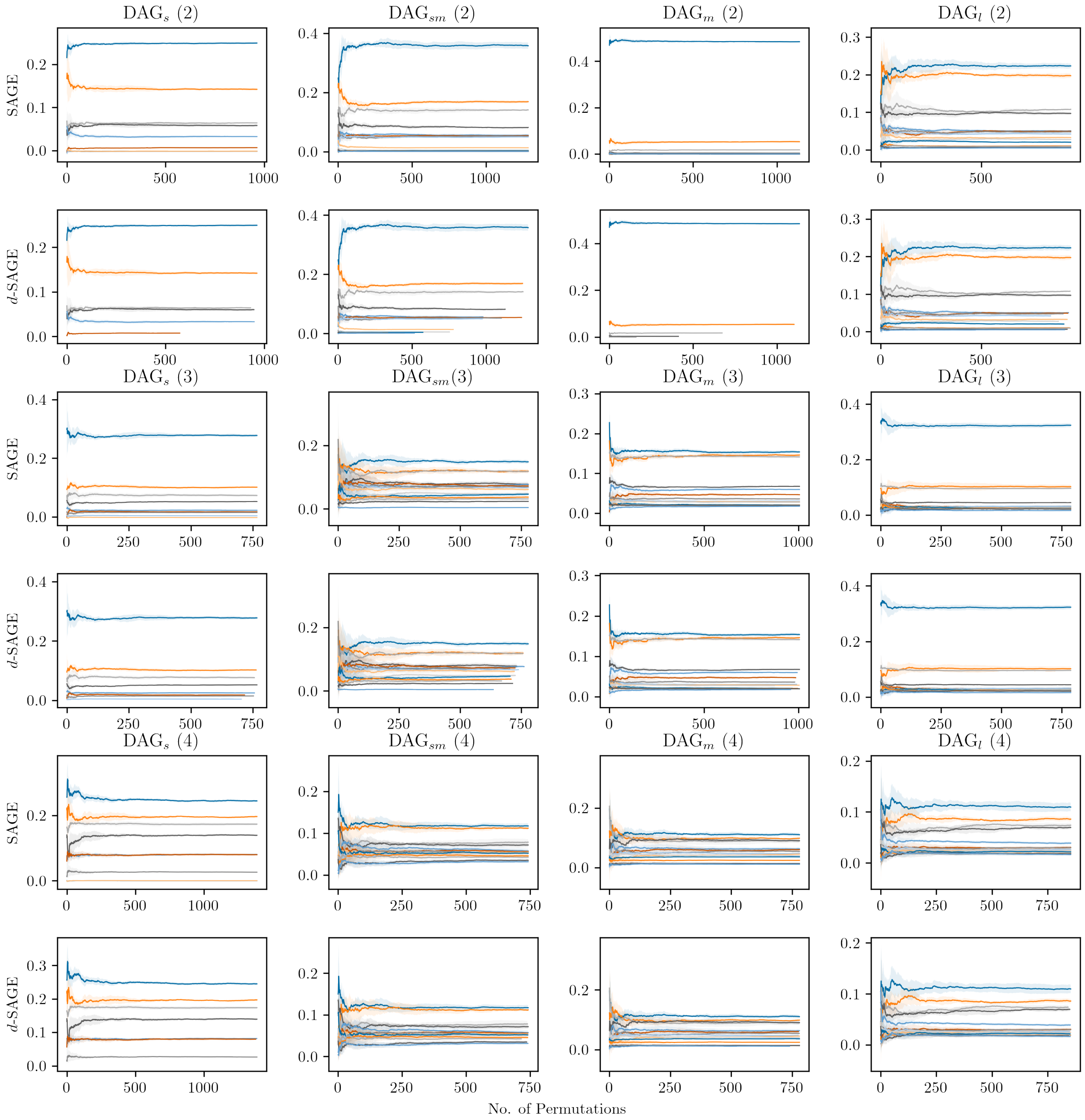


Figure 15: Convergence of largest fifteen SAGE and d-SAGE values for random forest models (RF) for every DAG (average adjacency degree). Each colour represents the same feature in SAGE and d-SAGE plots for a given graph (if present in both). Legend omitted for readability.

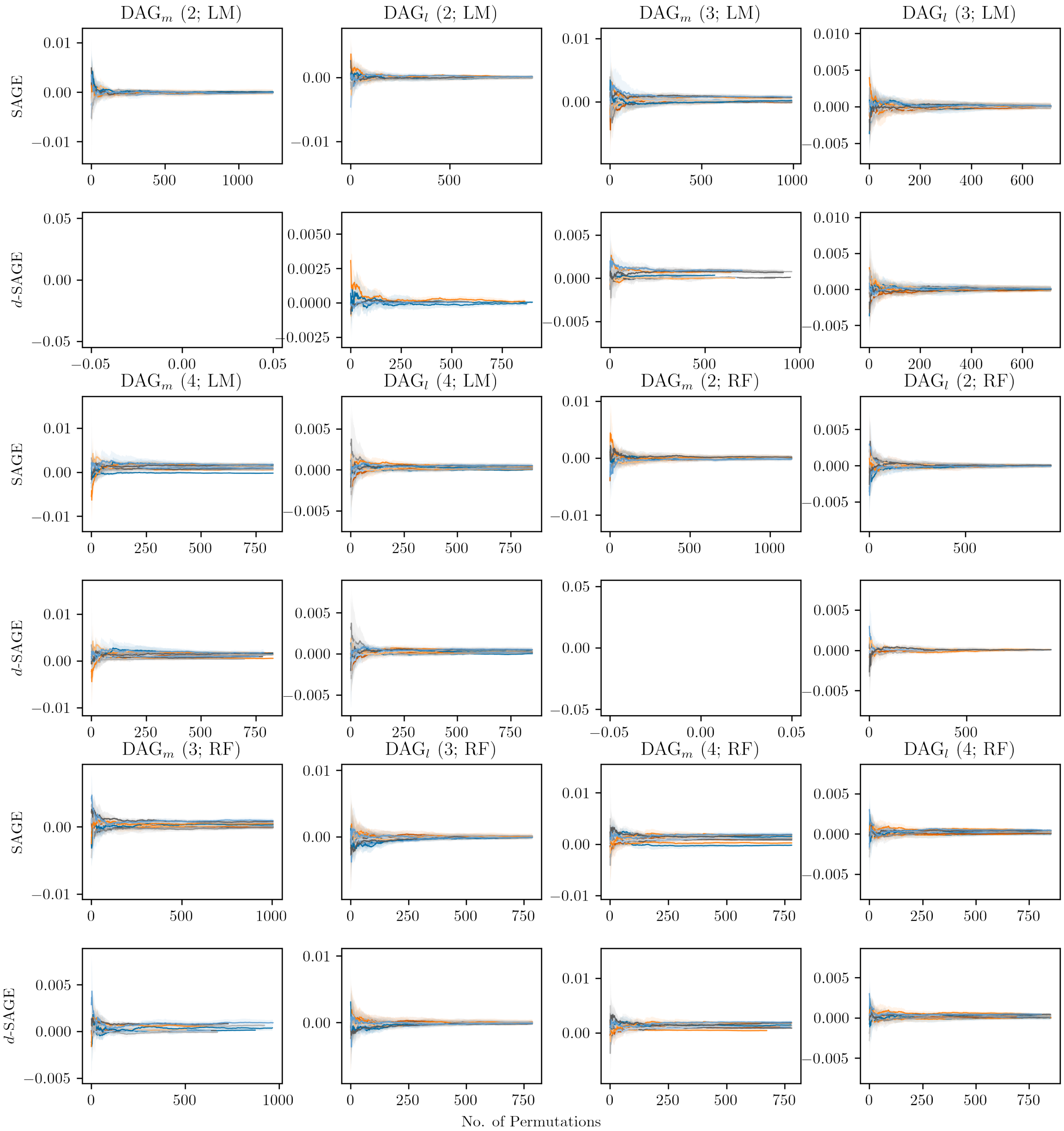


Figure 16: Convergence of bottom fifteen SAGE and d-SAGE values for optimal models (LM) and random forest models (RF) for DAG$_m$ and DAG$_l$ (average adjacency degree). Each colour represents the same feature in SAGE and d-SAGE plots for a given graph (if present in both). Legend omitted for readability.

E.1 Convergence of SAGE Values

The approximation algorithm is designed such that convergence for all values is required to stop. Hence, some values are converged but still computed. However, the benefit of d-SAGE depends on the share of CIs and not the number of permutations required for convergence, and hence, even a fewer number of permutations would lead to a similar speedup. Missing lines in the convergence plots belong to conditionally independent features (given every sampled coalition), which highlights the ability of (d-)SAGE for post-hoc feature selection. An example of faster converging d-SAGE values is displayed by the comparison of SAGE and d-SAGE for DAG$_m$(2) in Figure 14, where the small values (slightly above zero) converge faster for d-SAGE.

F PARTIAL CORRELATION TESTS v d-SEPARATION QUERIES

To highlight the benefit of CSL over statistical independence tests, we compared the runtime of linear time d-separation queries (in graphs inferred by TABU) from the NetworkX package for Python (Hagberg et al., 2008) to that of partial correlation tests for linear Gaussian data from the Pingouin package (Vallat, 2018). Results are based on 100 permutations. Table 4 clearly shows that partial correlation tests are typically more accurate at the cost of much higher runtime in comparison to d-separation queries (+ graph learning).

Table 4: Partial correlation tests v d-separation queries based on $n = 10,000$ and 100 permutations; Graph learning based on TABU; ACC = Accuracy.

DATA	TIME (d-separation)	TIME (TABU)	TIME (CIs)	ACC (d-separation)	ACC (CIs)
DAG$_s$ (2)	0.13s	0.06s	46.82s	1.000	1.000
DAG$_{sm}$ (2)	0.39s	0.22s	166.75s	0.996	0.999
DAG$_m$ (2)	1.95s	1.11s	1058.80s	1.000	1.000
DAG$_l$ (2)	15.48s	12.02s	4344.81s	0.863	0.934
DAG$_s$ (3)	0.13s	0.18s	47.15s	1.0	1.0
DAG$_{sm}$ (3)	0.41s	0.71s	166.28s	0.996	0.992
DAG$_m$ (3)	2.12s	2.51s	1089.00s	0.908	0.983
DAG$_l$ (3)	16.85s	18.22s	4299.47s	0.857	0.941
DAG$_s$ (4)	0.14s	0.09s	47.18s	1.0	0.998
DAG$_{sm}$ (4)	0.42s	1.37s	163.48s	1.0	0.988
DAG$_m$ (4)	2.33s	5.65s	1093.50s	0.845	0.940
DAG$_l$ (4)	20.74s	39.86s	4312.16s	0.902	0.916

Supplements for Paper VII: Model-agnostic Feature Importance and Effects with Dependent Features

Appendix A Decompose conditional PFI into cs-PFIs

Assuming a perfect construction of G_j, it holds that $X_j \perp X_{-j}|G_j$ and also that $X_j \perp G_j|X_{-j}$ (as G_j is a compression of X_{-j}). Therefore

$$P(X_j|X_{-j}) = P(X_j|X_{-j}, G_j) = P(X_j|G_j). \tag{8}$$

When we sample the replacement $\tilde{x}_j^{(i)}$for an $x_j^{(i)}$ from the marginal within a group ($P(X_j|G_j = g_j^{(i)})$, e.g., via permutation) we also sample from the conditional $P(X_j|X_{-j} = x_{-j}^{(i)})$. Every data point from the global sample can therefore equivalently be seen as a sample from the marginal within the group, or as a sample from the global conditional distribution.
As follows, the weighted sum of marginal subgroup PFIs coincides with the conditional PFI (cPFI).

$$cPFI = \sum_{i=1}^{n} \frac{1}{n} \left(L\left(f\left(\tilde{x}_j^{(i)}, x_{-j}^{(i)} \right), y^{(i)} \right) - L\left(\hat{f}\left(x_j^{(i)}, x_{-j}^{(i)} \right), y^{(i)} \right) \right) \tag{9}$$

$$= \sum_{k=1}^{K} \frac{n_k}{n} \sum_{i \in \mathcal{G}_k} \frac{1}{n_k} \left(L\left(f\left(\tilde{x}_j^{(i)}, x_{-j}^{(i)} \right), y^{(i)} \right) - L\left(\hat{f}\left(x_j^{(i)}, x_{-j}^{(i)} \right), y^{(i)} \right) \right) \tag{10}$$

$$= \sum_{k=1}^{K} \frac{n_k}{n} PFI^k \tag{11}$$

Appendix B Expectation and variance of the PFI in a subgroup

We show that under feature independence the PFI and a PFI in an arbitrary subgroup have the same expected value and the subgroup k PFI has a higher variance. Let $\tilde{L}^{(i)} = \frac{1}{M} \sum_{m=1}^{M} L(y^{(i)}, \hat{f}(\tilde{x}_{j,m}^{(i)}, x_{-j}^{(i)})$ and $L^{(i)} = L(y^{(i)}, \hat{f}(x_{j,m}^{(i)}, x_{-j}^{(i)})$.

Proof

$$\begin{aligned}
\mathbb{E}_{X_{-j}}[PFI_j] &= \mathbb{E}_{X_{-j}} \left[\frac{1}{n} \sum_{i=1}^{n} (\tilde{L}^{(i)} - L^{(i)}) \right] \\
&= \mathbb{E}_{X_{-j}}[\tilde{L}^{(i)} - L^{(i)}] \\
\mathbb{E}[PFI_j^k]_{X_{-j}} &= \mathbb{E}_{X_{-j}} \left[\frac{1}{n_k} \sum_{i:x^{(i)} \in \mathcal{G}_j^k} (\tilde{L}^{(i)} - L^{(i)}) \right]
\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{n_k}\mathbb{E}_{X_{-j}}\left[\sum_{i:x^{(i)}\in\mathcal{G}_j^k}(\tilde{L}^{(i)} - L^{(i)})\right]\\
&= \frac{1}{n_k}n_k\mathbb{E}_{X_{-j}}\left[(\tilde{L}^{(i)} - L^{(i)})\right]\\
&= \mathbb{E}_{X_{-j}}[PFI_j]\\
\mathbb{V}_{X_{-j}}\left[PFI_j\right] &= \mathbb{V}_{X_{-j}}\left[\frac{1}{n}\sum_{i=1}^{n}(\tilde{L}^{(i)} - L^{(i)})\right]\\
&= \frac{1}{n^2}n\mathbb{V}_{X_{-j}}\left[\tilde{L}^{(i)} - L^{(i)}\right]\\
&= \frac{1}{n}\mathbb{V}_{X_{-j}}\left[\tilde{L}^{(i)} - L^{(i)})\right]\\
\mathbb{V}_{X_{-j}}\left[PFI_j^k]\right] &= \mathbb{V}_{X_{-j}}\left[\frac{1}{n_k}\sum_{i=1}^{n_k}(\tilde{L}^{(i)} - L^{(i)})\right]\\
&= \frac{1}{n_k^2}n_k\mathbb{V}_{X_{-j}}\left[\tilde{L}^{(i)} - L^{(i)}\right]\\
&= \frac{1}{n_k}\mathbb{V}_{X_{-j}}\left[\tilde{L}^{(i)} - L^{(i)})\right]\\
\frac{\mathbb{V}_{X_{-j}}[PFI_j^k]}{\mathbb{V}_{X_{-j}}\left[PFI_j\right]} &= \frac{n}{n_k}
\end{aligned}
$$

□

Appendix C Expectation and variance of the PDP in a subgroup

We show that under feature independence the PDP and a PDP in an arbitrary subgroup have the same expected value and the subgroup k PDP has a higher variance.

Proof

$$
\begin{aligned}
\mathbb{E}_{X_{-j}}[PDP_j(x)] &= \mathbb{E}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]\\
\mathbb{E}_{X_{-j}}[PDP_j^k(x)] &= \mathbb{E}_{X_{-j}}\left[\frac{1}{n_k}\sum_{i=1}^{n_k}\hat{f}\big(x, x_{-j}^{(i)})\big)\right] = \frac{1}{n_k}n_k\mathbb{E}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]\\
&= \mathbb{E}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]\\
\mathbb{V}_{X_{-j}}\left[PDP_j(x)\right] &= \mathbb{V}_{X_{-j}}\left[\frac{1}{n}\sum_{i=1}^{n}\hat{f}\big(x, x_{-j}^{(i)}\big)\right]\\
&= \frac{1}{n^2}n\mathbb{V}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{n}\mathbb{V}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]\\
\mathbb{V}_{X_{-j}}\left[PDP_j^k(x)\right] &= \mathbb{V}_{X_{-j}}\left[\frac{1}{n_k}\sum_{i=1}^{n_k}\hat{f}\left(x, x_{-j}^{(i)}\right)\right]\\
&= \frac{1}{n_k^2}n_{k_j}\mathbb{V}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]\\
&= \frac{1}{n_k}\mathbb{V}_{X_{-j}}\left[\hat{f}(x, X_{-j})\right]\\
\frac{\mathbb{V}_{X_{-j}}[PDP_j^k(x)]}{\mathbb{V}_{X_{-j}}\left[PDP_j(x)\right]} &= \frac{n}{n_k}
\end{aligned}
$$

□

Appendix D cPFI ground truth scenario II

This chapter contains the results for the conditional PFI ground truth simulation, scenario II with an intermediate random forest (Table 6 and Fig. 11).

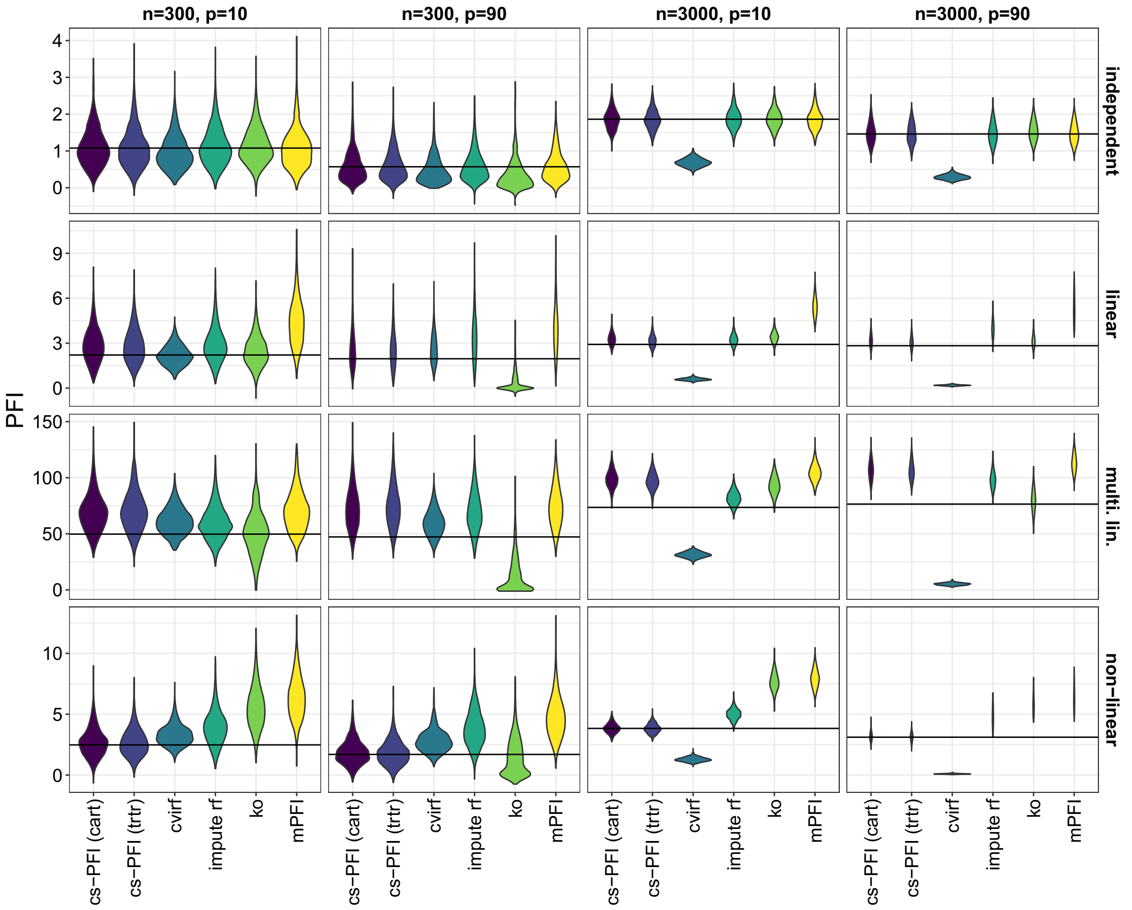


Fig. 11 Experiment (II) comparing various conditional PFI approaches with an intermediary a random forest against the true conditional PFI based on the data generating process

Table 6 MSE comparing estimated and true conditional PFI (for random forest, scenario II)

Setting	cs-PFI (cart)	cs-PFI (trtr)	cvirf	impute rf	ko	mPFI
independent						
n = 300, p = 10	0.26	0.28	0.22	0.27	0.25	0.27
n = 300, p = 90	0.19	0.17	0.14	0.18	0.19	0.17
n = 3000, p = 10	0.07	0.07	1.39	0.07	0.06	0.08
n = 3000, p = 90	0.08	0.08	1.37	0.08	0.08	0.08
linear						
n = 300, p = 10	1.79	1.69	0.45	1.87	1.10	7.11
n = 300, p = 90	1.93	1.88	1.36	4.25	2.93	7.06
n = 3000, p = 10	0.29	0.22	5.41	0.25	0.40	6.80
n = 3000, p = 90	0.32	0.24	6.98	1.66	0.26	7.02
multi. lin.						
n = 300, p = 10	667.79	744.48	275.58	335.40	377.35	726.15
n = 300, p = 90	972.42	1098.74	301.26	823.89	1473.67	1065.26
n = 3000, p = 10	715.41	625.99	1790.45	114.71	454.26	1017.53
n = 3000, p = 90	974.37	945.19	5090.09	532.44	110.94	1416.30
non-linear						
n = 300, p = 10	1.40	1.29	1.37	3.96	12.35	18.51
n = 300, p = 90	1.06	1.03	2.05	6.77	2.38	12.32
n = 3000, p = 10	0.17	0.16	6.53	1.55	15.29	17.56
n = 3000, p = 90	0.15	0.14	9.09	3.28	8.00	11.30

impute rf: Imputation with a random forest, ko: Model-X knockoffs, mPFI: (marginal) PFI, tree cart: cs-permutation based on CART, tree trtr: cs-permutation based on transformation trees, CVIRF: conditional variable importance for random forests

Appendix E cPFI ground truth tree depth

See Fig. 12.

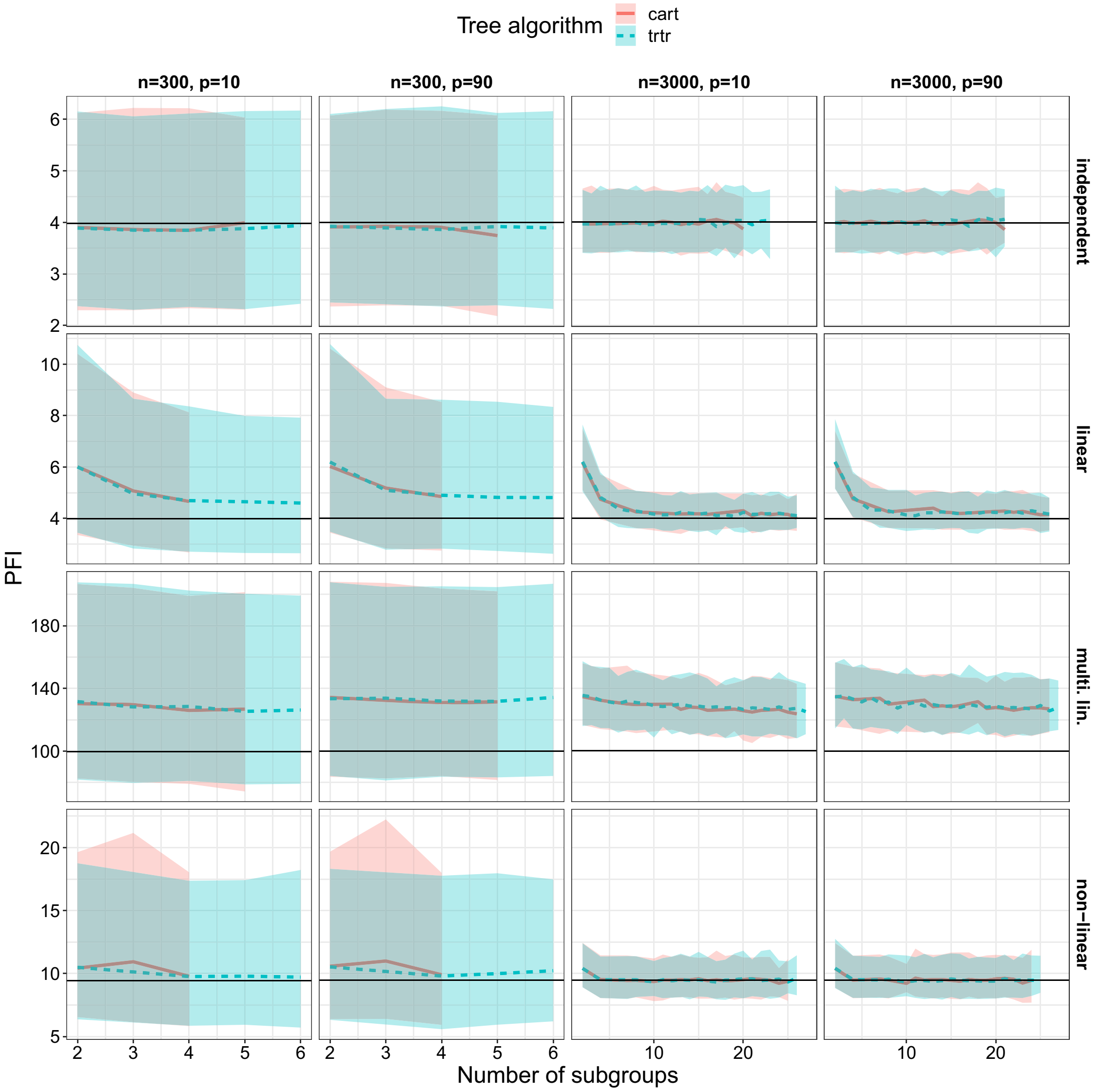


Fig. 12 Conditional PFI estimate using cs-PFI (**cart/tr**ansformation **tr**ee) with increasing number of subgroups (simulation scenario I). Displayed is the median PFI over 1000 repetitions along with the 5% and 95% quartiles

Appendix F Data fidelity on OpenML-CC18 data sets

An overview of data sets from the OpenML-CC18 benchmarking suit. We used a subset of 42 out of 72 data sets with 7 to 500 continuous features (Table 7).

Table 7 Overview of OpenML CC18 data sets used for the data fidelity experiment

OpenML ID	Name	No. Obs.	No. numerical feat.	No. feat.
1049	pc4	1458	38	38
1050	pc3	1563	38	38
1053	jm1	10,880	22	22
1063	kc2	522	22	22
1067	kc1	2109	22	22
1068	pc1	1109	22	22
12	mfeat-factors	2000	217	217
14	mfeat-fourier	2000	77	77
1461	bank-marketing	45,211	8	17
1475	first-order-theorem-proving	6118	52	52
1480	ilpd	583	10	11
1486	nomao	34,465	90	119
1487	ozone-level-8hr	2534	73	73
1494	qsar-biodeg	1055	42	42
1497	wall-robot-navigation	5456	25	25
15	breast-w	683	10	10
1501	semeion	1593	257	257
151	electricity	45,312	8	9
1510	wdbc	569	31	31
16	mfeat-karhunen	2000	65	65
182	satimage	6430	37	37
188	eucalyptus	641	15	20
22	mfeat-zernike	2000	48	48
23517	numerai28.6	96,320	22	22
28	optdigits	5620	63	65
307	vowel	990	11	13
31	credit-g	1000	8	21
32	pendigits	10,992	17	17
37	diabetes	768	9	9
40499	texture	5500	41	41
40701	churn	5000	17	21
40966	MiceProtein	552	78	82
40979	mfeat-pixel	2000	241	241
40982	steel-plates-fault	1941	28	28
40984	segment	2310	19	20
40994	climate-model-simulation-crashes	540	21	21
44	spambase	4601	58	58
4538	GesturePhaseSegmentationProcessed	9873	33	33
458	analcatdata_authorship	841	71	71
54	vehicle	846	19	19
6	letter	20,000	17	17
6332	cylinder-bands	378	19	40

Appendix F.1 Data fidelity results

See Figs. 13, 14, 15 and 16.

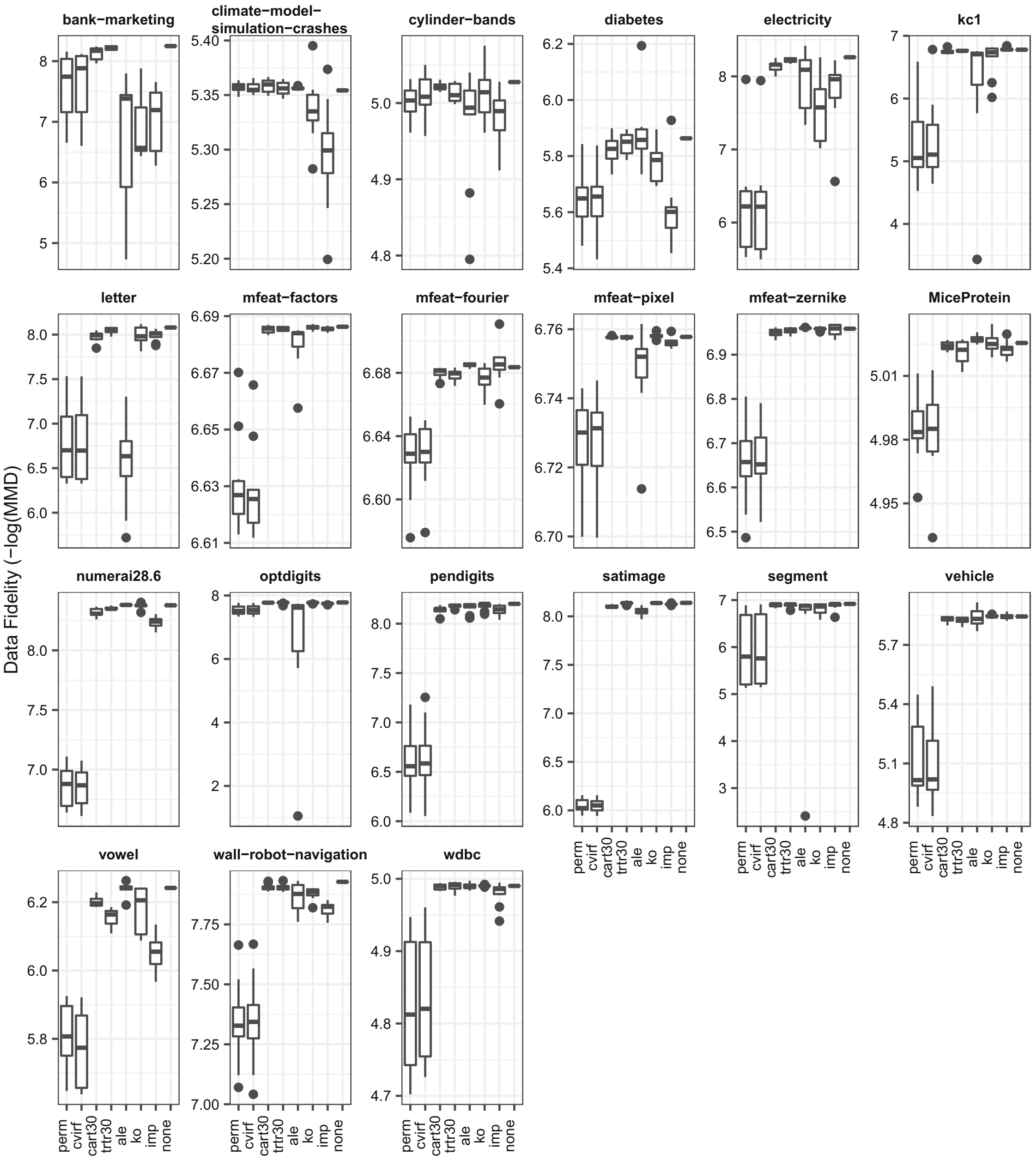


Fig. 13 Data Fidelity experiment with OpenML-CC18 data sets (1/2). Different sampling types are compared: unconditional permutation (perm), cs-permutation (maximal tree depth) with CART (cart30) or transformation trees (trtr30), Model-X knockoffs (ko), data imputation with a random forest (imp), ALE (ale), conditional variable importance for random forests (cvirf) and no permutation (none). Each data point in the boxplot represents one feature and one data set. Results from repeated experiments have been averaged (mean) before using them in the boxplots

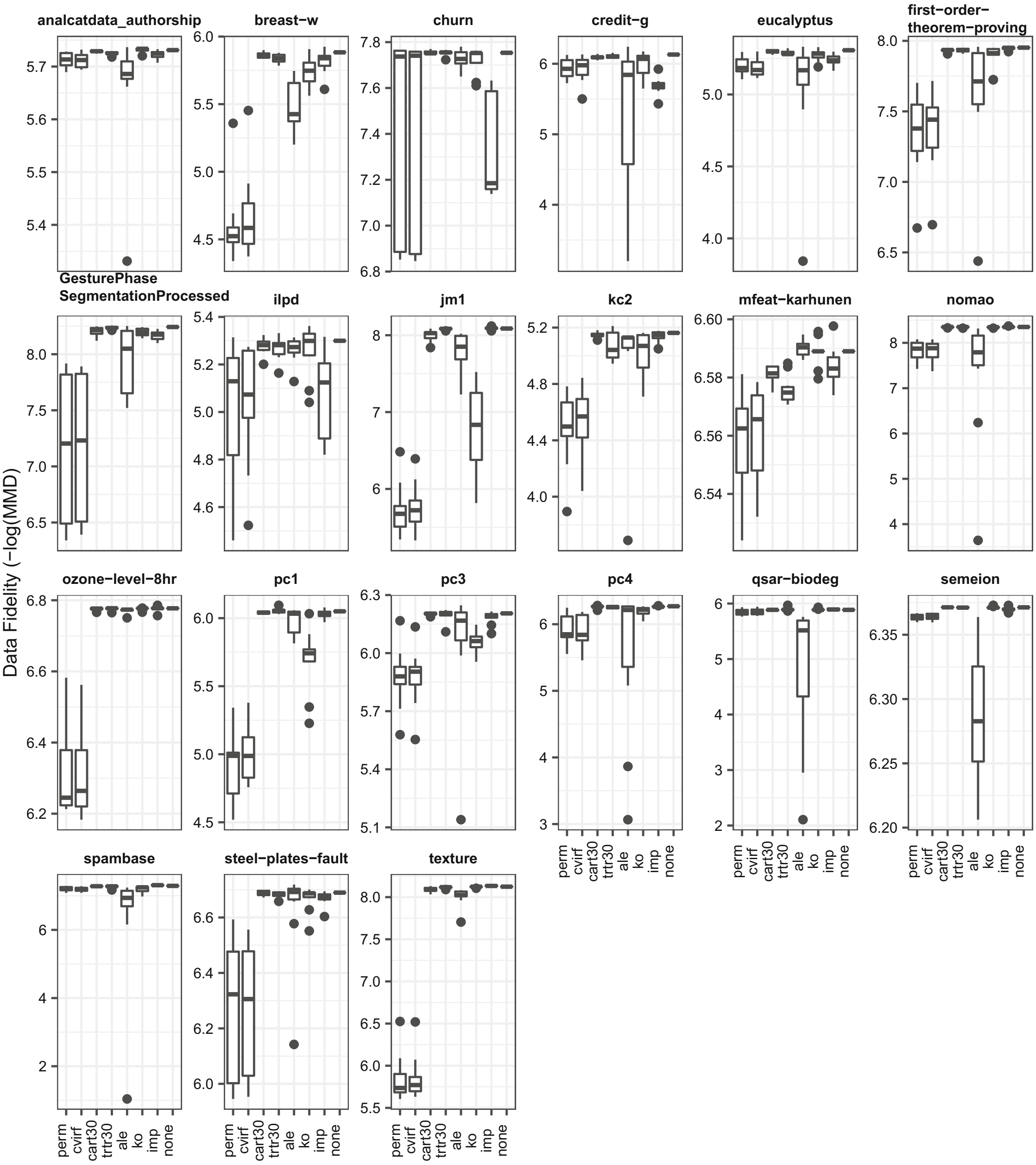


Fig. 14 Data Fidelity experiment with OpenML-CC18 data sets (1/2). Different sampling types are compared: unconditional permutation (perm), cs-permutation (maximal tree depth) with CART (cart30) or transformation trees (trtr30), Model-X knockoffs (ko), data imputation with a random forest (imp), ALE (ale), conditional variable importance for random forests (cvirf) and no permutation (none). Each data point in the boxplot represents one feature and one data set. Results from repeated experiments have been averaged (mean) before using them in the boxplots

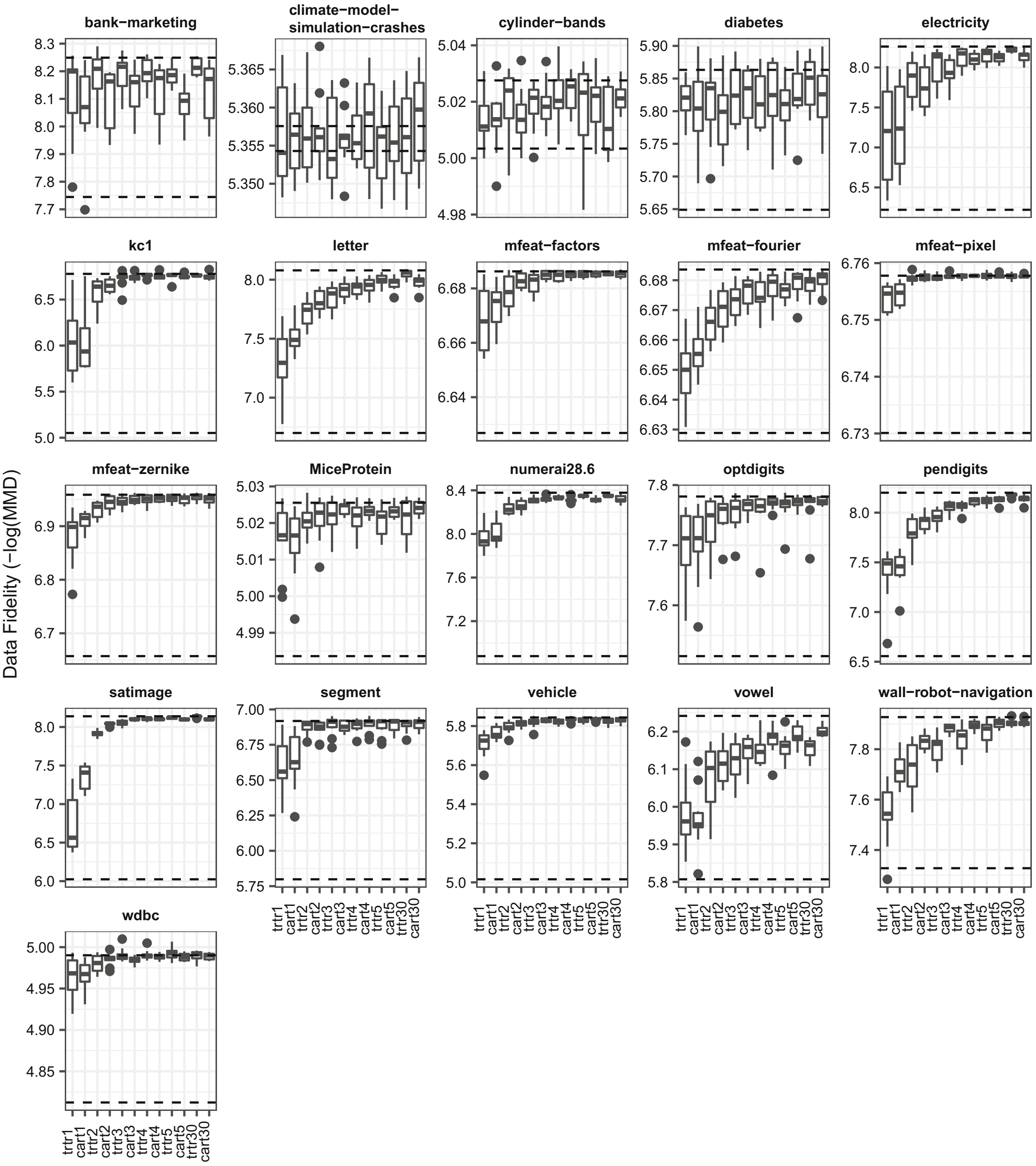


Fig. 15 Data Fidelity experiment with OpenML-CC18 data sets (1/2). Different tree depths and tree types (CART and Transformation Trees) are compared. Unconditional permutation and lack of permutation serve as lower and upper bound for data fidelity and their median data fidelity is plotted as dotted lines. Each data point in the boxplot represents one feature and one data set. Results from repeated experiments have been averaged (mean) before using them in the boxplots

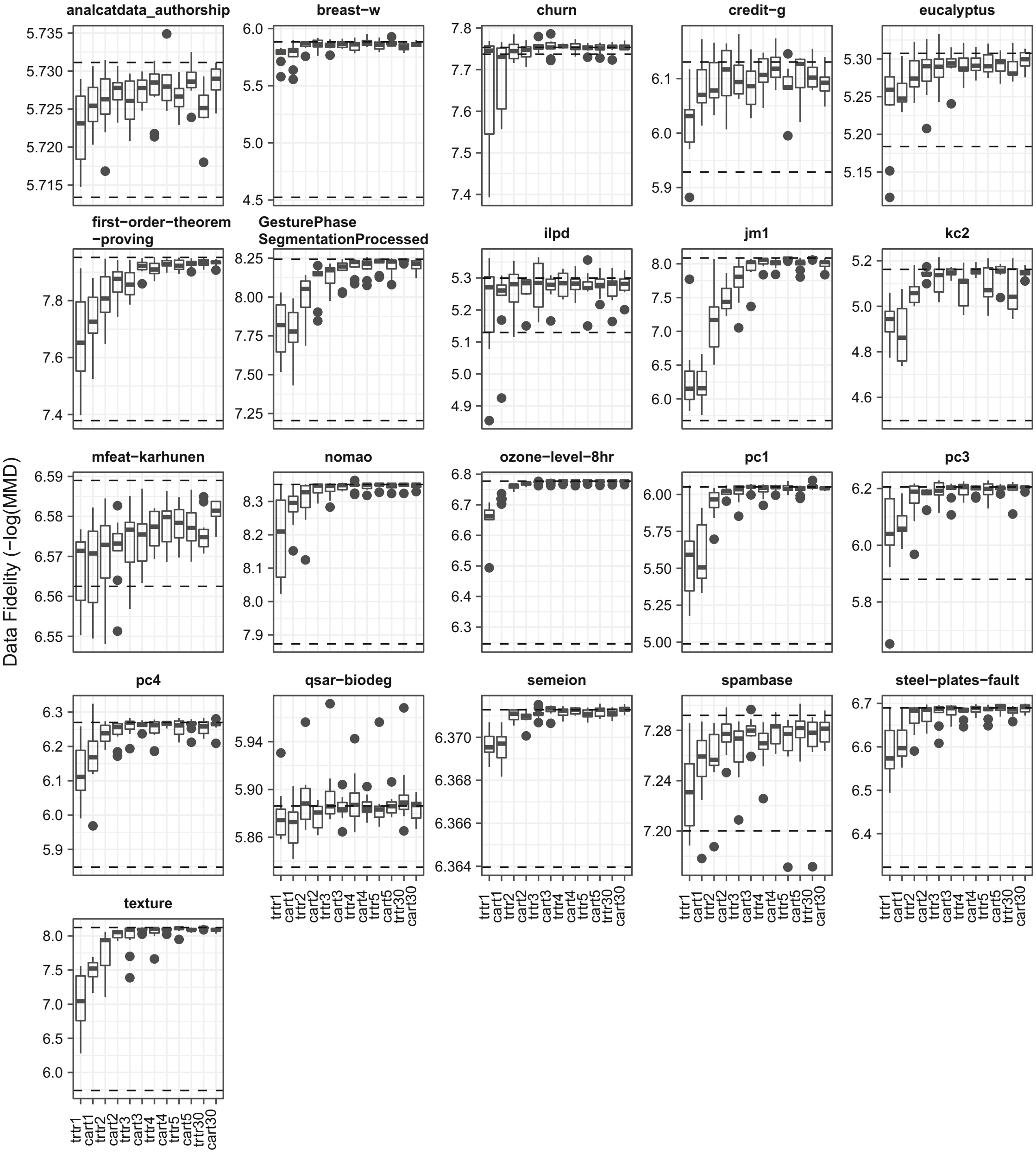


Fig. 16 Data Fidelity experiment with OpenML-CC18 data sets (1/2). Different tree depths and tree types (CART and Transformation Trees) are compared. Unconditional permutation and lack of permutation serve as lower and upper bound for data fidelity and their median data fidelity is plotted as dotted lines. Each data point in the boxplot represents one feature and one data set. Results from repeated experiments have been averaged (mean) before using them in the boxplots

Appendix G Model fidelity plots

See Fig. 17.

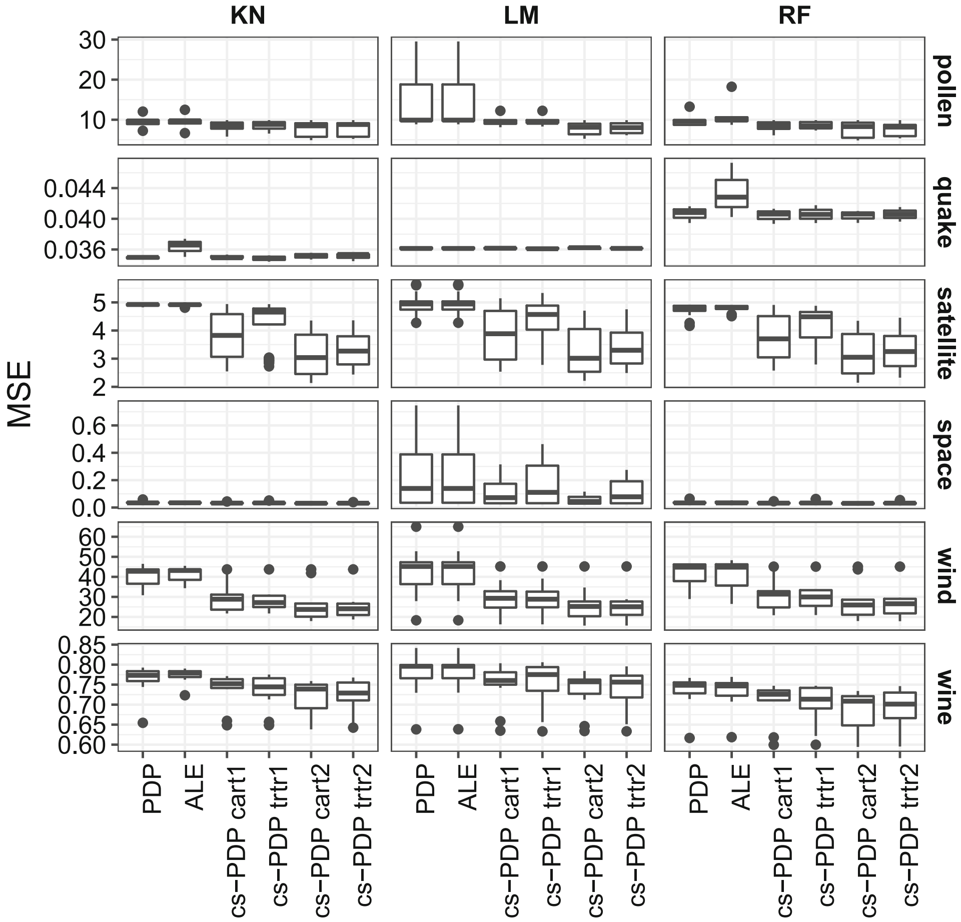


Fig. 17 Comparing the loss between model f and various feature effect methods. Each instance in the boxplot is MSE for one feature, summed over the test data

Appendix H Application: feature dependence analysis

The features in the bike data are dependent. For example, the correlation between temperature and humidity is 0.13. The data contains both categorical and numerical features and we are interested in the multivariate, non-linear dependencies. Thus, correlation is an inadequate measure of dependence. We therefore indicate the degree of dependence by showing the extent to which we can predict each feature from all other features in Table 8. This idea is based on the proportional reduction in loss (Cooil and Rust 1994). Per feature, we trained a random forest to predict that feature from all other features. We measured the proportion of loss explained by each random forest, compared to a constant model to quantify the dependence of the respective feature on all other features. For numerical features, this meant using the R-squared measure. For categorical features, we computed $1 - MMCE(y_{class}, rf(X))/MMCE(y_{class}, x_{mode})$, where $MMCE$ is the mean misclassification error, y_{class} the true class, $rf()$ the classification function of the random forest and x_{mode} the most frequent class in the training data. We divided the training data into two folds and trained the random forest on one half. Then, we computed the proportion of explained loss on the other half and vice versa. Finally, we averaged the results. The feature "work" can be fully predicted by weekday and holiday. Season, temperature, humidity and weather can be partially predicted and are therefore not independent.

Table 8 Percentage of loss explained by predicting a feature from the remaining features with a random forest

Season	Holiday	Weekday	Temp	Hum	Work	Weather	Year	Wind
46%	25%	12%	66%	42%	100%	44%	10%	11%

References

Apley DW, Zhu J (2016) Visualizing the effects of predictor variables in black box supervised learning models. arXiv preprint arXiv:1612.08468

Bair E, Ohrbach R, Fillingim RB, Greenspan JD, Dubner R, Diatchenko L, Helgeson E, Knott C, Maixner W, Slade GD (2013) Multivariable modeling of phenotypic risk factors for first-onset TMD: the OPPERA prospective cohort study. J Pain 14(12):T102–T115

Bischl B, Casalicchio G, Feurer M, Hutter F, Lang M, Mantovani RG, van Rijn JN, Vanschoren J (2019) Openml benchmarking suites. arXiv preprint arXiv:1708.03731

Boulesteix AL, Wright MN, Hoffmann S, König IR (2020) Statistical learning approaches in the genetic epidemiology of complex diseases. Hum Genet 139(1):73–84

Breiman L (2001) Random forests. Mach Learn 45(1):5–32

Breiman L, Friedman J, Olshen R, Stone C (1984) Classification and regression trees. Wadsworth and Brooks, Boston

Bryk AS, Raudenbush SW (1992) Hierarchical linear models: applications and data analysis methods. Sage Publications Inc, Thousand Oaks

Candes E, Fan Y, Janson L, Lv J (2018) Panning for gold:'model-X'knockoffs for high dimensional controlled variable selection. J R Stat Soc Ser B (Stat Methodol) 80(3):551–577

Casalicchio G, Bossek J, Lang M, Kirchhoff D, Kerschke P, Hofner B, Seibold H, Vanschoren J, Bischl B (2017) OpenML: an R package to connect to the machine learning platform OpenML. Comput Stat 34:977–991

Chen H, Janizek JD, Lundberg S, Lee SI (2020) True to the model or true to the data? arXiv preprint arXiv:2006.16234

Cooil B, Rust RT (1994) Reliability and expected loss: a unifying principle. Psychometrika 59(2):203–216

Debeer D, Strobl C (2020) Conditional permutation importance revisited. BMC Bioinform 21(1):1–30

Dua D, Graff C (2017) UCI machine learning repository. http://archive.ics.uci.edu/ml

Esselman PC, Stevenson RJ, Lupi F, Riseng CM, Wiley MJ (2015) Landscape prediction and mapping of game fish biomass, an ecosystem service of Michigan rivers. N Am J Fish Manag 35(2):302–320

Fisher A, Rudin C, Dominici F (2019) All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously. J Mach Learn Res 20(177):1–81

Fortet R, Mourier E (1953) Convergence de la répartition empirique vers la répartition théorique. Ann Sci l'École Normale Supér 70:267–285

Freiesleben T, König G, Molnar C, Tejero-Cantero A (2022) Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena. arXiv preprint arXiv:2206.05487

Friedman JH et al (1991) Multivariate adaptive regression splines. Ann Stat 19(1):1–67

Frye C, de Mijolla D, Begley T, Cowton L, Stanley M, Feige I (2020) Shapley explainability on the data manifold. arXiv preprint arXiv:2006.01272

Goldstein A, Kapelner A, Bleich J, Pitkin E (2015) Peeking inside the black box: visualizing statistical learning with plots of individual conditional expectation. J Comput Graph Stat 24(1):44–65

Gregorutti B, Michel B, Saint-Pierre P (2017) Correlation and variable importance in random forests. Stat Comput 27(3):659–678

Gretton A, Fukumizu K, Teo CH, Song L, Schölkopf B, Smola AJ et al (2007) A kernel statistical test of independence. Nips Citeseer 20:585–592

Gretton A, Borgwardt KM, Rasch MJ, Schölkopf B, Smola A (2012) A kernel two-sample test. J Mach Learn Res 13(1):723–773

Hooker G (2007) Generalized functional anova diagnostics for high-dimensional functions of dependent variables. J Comput Graph Stat 16(3):709–732

Hooker G, Mentch L (2019) Please stop permuting features: an explanation and alternatives. arXiv preprint arXiv:1905.03151

Hothorn T (2018) Top-down transformation choice. Stat Model 18(3–4):274–298

Hothorn T, Zeileis A (2015) partykit: a modular toolkit for recursive partytioning in R. J Mach Learn Res 16(1):3905–3909

Hothorn T, Zeileis A (2017) Transformation forests. arXiv preprint arXiv:1701.02110

König G, Molnar C, Bischl B, Grosse-Wentrup M (2020) Relative feature importance. arXiv preprint arXiv:2007.08283

Lang M, Binder M, Richter J, Schratz P, Pfisterer F, Coors S, Au Q, Casalicchio G, Kotthoff L, Bischl B (2019) mlr3: a modern object-oriented machine learning framework in R. J Open Source Softw 4:1903

Lei J, G'Sell M, Rinaldo A, Tibshirani RJ, Wasserman L (2018) Distribution-free predictive inference for regression. J Am Stat Assoc 113(523):1094–1111

Molnar C, Bischl B, Casalicchio G (2018) iml: an R package for interpretable machine learning. JOSS 3(26):786

Obringer R, Nateghi R (2018) Predicting urban reservoir levels using statistical learning techniques. Sci Rep 8(1):1–9

Parr T, Wilson JD (2019) A stratification approach to partial dependence for codependent variables. arXiv preprint arXiv:1907.06698

Patterson E, Sesia M (2020) knockoff: the knockoff filter for controlled variable selection. R package version 0.3.3. https://CRAN.R-project.org/package=knockoff

Pintelas E, Liaskos M, Livieris IE, Kotsiantis S, Pintelas P (2020) Explainable machine learning framework for image classification problems: case study on glioma cancer prediction. J Imaging 6(6):37

R Core Team (2017) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna

Ribeiro MT, Singh S, Guestrin C (2016) Why should I trust you?: Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, ACM, pp 1135–1144

Romano Y, Sesia M, Candès E (2019) Deep knockoffs. J Am Stat Assoc, pp 1–12

Scholbeck CA, Molnar C, Heumann C, Bischl B, Casalicchio G (2019) Sampling, intervention, prediction, aggregation: a generalized framework for model-agnostic interpretations. In: Joint European conference on machine learning and knowledge discovery in databases. Springer, pp 205–216

Smola A, Gretton A, Song L, Schölkopf B (2007) A Hilbert space embedding for distributions. In: International conference on algorithmic learning theory. Springer, pp 13–31

Stachl C, Au Q, Schoedel R, Gosling SD, Harari GM, Buschek D, Völkel ST, Schuwerk T, Oldemeier M, Ullmann T, Hussmann H, Bischl B, Bühner M (2020) Predicting personality from patterns of behavior collected with smartphones. Proc Natl Acad Sci 117(30):17680–17687

Stiglic G, Kocbek P, Fijacko N, Zitnik M, Verbert K, Cilar L (2020) Interpretability of machine learning-based prediction models in healthcare. Wiley Interdiscip Rev Data Min Knowl Discov 10(5):e1379

Strobl C, Boulesteix AL, Kneib T, Augustin T, Zeileis A (2008) Conditional variable importance for random forests. BMC Bioinform 9(1):307

Szepannek G (2019) How much can we see? A note on quantifying explainability of machine learning models. arXiv preprint arXiv:1910.13376

Vanschoren J, Van Rijn JN, Bischl B, Torgo L (2014) OpenML: networked science in machine learning. ACM SIGKDD Explor Newsl 15(2):49–60

Watson DS, Wright MN (2021) Testing conditional independence in supervised learning algorithms. Mach Learn 110(8):2107–2129

Zhao X, Yan X, Yu A, Van Hentenryck P (2020) Prediction and behavioral analysis of travel mode choice: a comparison of machine learning and logit models. Travel Behav Soc 20:22–35

Publication List

König, Gunnar, Timo Freiesleben, and Moritz Grosse-Wentrup. **Improvement-Focused Causal Recourse (ICR).** *Proceedings of the AAAI Conference on Artificial Intelligence.* Vol. 37. No. 10. 2023.

Gunnar König contributed to the paper as first author. Gunnar König had the initial idea, wrote large parts of the paper, developed the proofs and wrote the code. Timo Freiesleben helped to develop the story and the philosophical foundation, wrote large parts of Section 4, checked the proofs and contributed to Sections 1, 2, 9 and 10. All authors helped to revise and proofread the paper.

For legal reasons the arXiv preprint of the article is included in this document (https://doi.org/10.48550/arXiv.2210.15709). The original version can be accessed via https://doi.org/10.1609/aaai.v37i10.26398.

Freiesleben, Timo, Gunnar König, Christoph Molnar, and Alvaro Tejero-Cantero. **Scientific Inference With Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena.** *arXiv preprint arXiv:2206.05487 (2022).*

Gunnar König contributed to the paper as co-author with significant contributions. Gunnar König wrote large parts of the section on causal learning. Gunnar König, Alvaro Tejero-Cantero, and Christoph Molnar added valuable new ideas, proofread and helped revise the paper. Timo Freiesleben wrote large parts of the paper and developed the initial idea. Alvara Tejero-Cantero helped design Figures 1,3,4, and 7 and contributed a paragraph on mechanistic models.

Molnar*, Christoph, Timo Freiesleben*, Gunnar König*, Julia Herbinger, Tim Reisinger, Giuseppe Casalicchio, Marvin Wright, and Bernd Bischl. **Relating the Partial Dependence Plot and Permutation Feature Importance to the Data Generating Process.** *The 1st World Conference on eXplainable Artificial Intelligence (2023).*

Gunnar König contributed to the paper as shared first author. Gunnar König and Timo Freiesleben led the revision of the paper and made conceptual contributions, particularly in Section 2.5, and contributed some of the proofs, particularly Theorem 3. Gunnar König implemented the application example and wrote sections 1.1 and 4. Christoph Molnar developed the initial idea and wrote large parts of the paper. Christoph Molnar and Marvin Wright implemented and ran the simulation study in Section 3.1; Julia Herbinger and Tim Reisinger implemented and ran the comparison with model-based approaches in Section 3.2. All authors added valuable new discussion points and helped revise the text.

König, Gunnar, Christoph Molnar, Bernd Bischl and Moritz Grosse-Wentrup. **Relative Feature Importance (RFI).** *2020 25th International Conference on Pattern Recognition (ICPR).* IEEE, 2021.

Gunnar König contributed to the paper as first author. Gunnar König had the initial idea and wrote large parts of the paper. Christoph Molnar partly wrote the section on estimation and testing, reviewed the software code, and extensively proofread the mathematical proofs. All authors added input, suggested modifications proofread and revised the paper.

Molnar, Christoph, Gunnar König, Julia Herbinger, Timo Freiesleben, Susanne Dandl, Christian Scholbeck, Giuseppe Casalicchio, Moritz Grosse-Wentrup and Bernd Bischl (2022). **General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models.** *International Workshop on Extending Explainable AI Beyond Deep Models and Classifiers.* Cham: Springer International Publishing, 2020.

Gunnar König contributed to the paper as a co-author with significant contributions. Gunnar König suggested the framing of the paper and wrote large parts of Sections 2, 5, and 10. Christoph Molnar initiated and coordinated the project. The co-authors mainly wrote the remaining sections. All authors added input to the chapters in which they were not involved as authors and proofread and revised the paper.

Luther*, Christoph, Gunnar König*, Moritz Grosse-Wentrup. **Efficient SAGE Estimation via Causal Structure Learning.** *International Conference on Artificial Intelligence and Statistics.* PMLR, 2023.

Gunnar König contributed to the paper as shared first author. Gunnar König had the initial idea, proved Theorem 1, wrote the feature importance code, did the real-world application and supervised Christoph Luther during his Master's thesis. Gunnar König and Christoph designed the experiments together. Christoph Luther conducted the experiments, visualized the results and wrote large parts of the paper. Gunnar König revised the structure of the article, contributing paragraphs to all sections. All authors helped to edit and proofread the paper.

Molnar, Christoph, Gunnar König, Bernd Bischl, Giuseppe Casalicchio. **Model-agnostic Feature Importance and Effects with Dependent Features: A Conditional Subgroup Approach.** *Data Mining and Knowledge Discovery (2023):* 1-39.

Gunnar König contributed to the paper as co-author with significant contributions. Gunnar Koenig wrote large parts of Section 1 and Section 4 and provided the proofs in Appendix A and B. Christoph Molnar wrote most of the paper. All authors added input, suggested modifications proofread and revised the paper.

Eidesstattliche Versicherung / Affidavit

Hiermit erkläre ich an Eidesstatt, dass die Dissertation “If Interpretability is the Answer, What is the Question?” von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

I hereby confirm that the dissertation “If Interpretability Is the Answer, What is the Question?” is the result of my own work and that I did not rely on forbidden help.

München, den 24. Juli 2023
Munich, the 24th of July 2023

. . . .
Gunnar König

Acknowledgements

First and foremost, I would like to thank my supervisor Moritz for his trust and continuous support. I got a nudge when needed but was also given the freedom to explore and tinker, which I am very grateful for. Our meetings not only taught me how to approach research but also guided me on how to be a supervisor and how to approach academic life (and where to get good coffee, of course).

Furthermore, I would like to thank my co-supervisor Bernd, who supported me since I started my master's at LMU. Thank you for integrating me into the research ecosystem that you created over the years and for the honest feedback and advice.

A great thank you goes to Thomas and Zach for being interested in our work and for volunteering to be the second and third referees of this thesis.

The IML group at LMU was a constant source of inspiration. I would like to thank Giuseppe, Christian, Christoph, Fiona, Julia, Timo and Susanne. Our weekly calls kept me going during the most difficult lockdown weeks, and I already miss our twenty to twelve o'clock mensa dates. I could not have wished for a more supportive and fun peer group, and I am sure we will stay in touch!

Furthermore, I'd like to thank my colleagues in Vienna: Alex, Anja, Anita, Akshey, Chen, Christoph, Mauricio, Nike, Peter, Philipp, Sadiq, and Triggvi. I always felt welcome during my visits, and I learned a lot from you in our weekly tea talks – about research and how to graft an apple tree. If you happen to be in Munich, I'm happy to return the guest friendliness I always received!

I would also like to thank Christoph and Valik for their trust in choosing me as their supervisor and for everything I learned from you during our time together.

There are more people to thank at LMU than I can list in this acknowledgement. I would like to thank Mina for being the best office mate, Heidi for being a role model, Anil and Philipp for the coffee dates, Dominik for being Dominik, Lisa for the breaks at ERS, Hannah for the much-needed support in crafting graduation caps, Fabian for regularly making me laugh out loud

when opening Mattermost, Moritz for the great time we had when offering the causality seminar, Jann for always getting a dessert as well, Martin for helping me fight the printer, Brigitte for making me enjoy bureaucracy and Thomas for supporting me with your organisational foresight and your philosophical curiosity!

Furthermore, I'd like to thank the Palmi Cafe for the hazelnut affogato, Cafe Schneller for the pear and poppy cake, Kara Kazan for the best Kumpir and the U-Bahn baker for the emergency Börek. You should try them if you are in the area (but maybe skip the Börek)! Also, I would like to thank my bike and the Bavarian Alps for allowing me to eat all that sugar.

I would not have made it through the PhD without my friends and family.

Special thanks go to Max, Teddy, Malte, Claudi, Markus, Chris, Simon, Martin, Pauline, Clari, Ti, Nina and Nina, Timo, Christoph, Heidi, Susanne and the whole Wehrle-WG for keeping me nourished when deadlines approached faster than expected and for helping me be silly.

Last but not least, I want to thank my parents, Karin and Sigurd and my sisters, Clara and Lisa, without whose support I would had the opportunity to do a PhD. Also, I want to thank Sophie for her loving support and for her patience while waiting for me to fix that one last bug.