Bridging Gaps in Interpretable Machine Learning

Sensitivity Analysis, Marginal Effects, and Cluster Explanations

Christian Alexander Scholbeck

2024

Bridging Gaps in Interpretable Machine Learning

Sensitivity Analysis, Marginal Effects, and Cluster Explanations

Christian Alexander Scholbeck

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

eingereicht von
Christian Alexander Scholbeck
am 28.02.2024

Erster Berichterstatter: Prof. Dr. Christian Heumann
Zweiter Berichterstatter: Prof. Dr. Bernd Bischl
Dritter Berichterstatter: Prof. Andrew Bennett, Ph.D.

Tag der Disputation: 14.05.2024

Eidesstattliche Versicherung

(Siehe Promotionsordnung vom 12. Juli 2011, §8 Abs. 2 Nr. 5)

Hiermit versichere ich an Eides statt, dass die Dissertation von mir selbstständig und ohne unerlaubte Beihilfe angefertigt ist.

München, den 28.02.2024

Christian Alexander Scholbeck

Acknowledgments

I would like to express my gratitude to all those who have supported and advised me throughout my doctoral studies, including ...

- *... Christian Heumann and Bernd Bischl for supervising my dissertation, their academic guidance, and the freedom to pursue my own ideas.*
- *... Hoshin Gupta for hosting my visit to the University of Arizona and providing invaluable advice on my research on sensitivity analysis.*
- *... Andrew Bennett for his willingness to act as a third referee for my dissertation.*
- *... David Rügamer and Helmut Küchenhoff for their willingness to be part of the examination panel.*
- *... my colleagues at the Department of Statistics and, notably, in the research group on interpretable machine learning for our friendships and technical discussions.*
- *... my colleagues at the University of Arizona for welcoming me during my visit and the many insights into the earth sciences.*
- *... all my family and friends in Germany and abroad.*

Abstract

The interpretation of models is an integral part of machine learning that is referred to as interpretable machine learning (IML). This dissertation is comprised of six papers that bridge the gap between IML and efforts to gain an understanding of mathematical models in other domains.

In recent years, an upsurge of research in IML brought along many novel model-agnostic interpretation methods. Although they might seem loosely connected, the first paper presents a generalized framework of work stages (sampling, intervention, prediction, aggregation) for model-agnostic interpretation methods. The second paper illustrates many general pitfalls of this methodology, which can lead to erroneous interpretations if applied incorrectly. Furthermore, model-agnostic techniques, as well as explanations of the hyperparameter optimization process, are related to sensitivity analysis, an auxiliary methodology used to explain complex systems in many fields such as environmental modeling or engineering. The third paper provides an impetus for discussion on how IML can be seen as sensitivity analysis applied to machine learning, which integrates recent advances in IML into a larger body of research on how to explain complex systems.

The fourth paper bridges the gap between IML and interpretations in applied statistics. It presents a model-agnostic interpretation approach with forward marginal effects (FMEs) to interpret any predictive model on the local, regional, and global level: The FME indicates the change in prediction for a pre-specified change in feature values; a non-linearity measure provides additional diagnostic information on whether the FME is a sufficient local descriptor of the prediction function; and the conditional average marginal effect aggregates local model explanations while preserving fidelity to the underlying predictive model. In addition, the fifth paper introduces the R package `fmeffects`, the first software implementation of the theory surrounding FMEs.

In spite of the practical relevance of explaining clusters, research in IML almost exclusively focuses on supervised learning. The sixth paper bridges the gap with cluster explanations. It introduces a framework of work stages (sampling, intervention, reassignment, aggregation) to design algorithm-agnostic cluster explanation methods termed FACT (feature attributions for clustering). Furthermore, it introduces two novel interpretation methods: SMART (scoring metric after permutation) measures changes in cluster assignments by custom scoring functions after permuting selected features; IDEA (isolated effect on assignment) indicates local and global changes in cluster assignments after making uniform changes to selected features.

Kurzfassung

Die Interpretation von Modellen ist ein integraler Bestandteil des maschinellen Lernens, der als interpretierbares maschinelles Lernen (IML) bezeichnet wird. Die vorliegende Dissertation besteht aus sechs Fachpublikationen, die Konzepte aus dem IML und der Interpretation mathematischer Modelle in anderen Fachgebieten verbinden.

In den vergangenen Jahren brachte ein Aufschung der Forschung viele neuartige modell-agnostische Interpretationsmethoden mit sich. Obwohl diese nur schwach verwandt erscheinen, stellt die erste Publikation ein allgemeines Konzept von Arbeitsschritten (Sampling, Intervention, Prediction, Aggregation) für modell-agnostische Interpretationsmethoden vor. Die zweite Publikation illustriert diverse allgemeine Fallstricke dieser Methodik, die bei unsachgemäßer Anwendung zu fälschlichen Interpretationen führt. Sowohl bei modell-agnostischen Verfahren als auch bei der Erklärung der Hyperparameteroptimierung findet sich eine Ähnlichkeit mit der Sensitivitätsanalyse, eine Hilfsmethodik zur Erklärung komplexer Systeme, die in vielen Fachgebieten wie den Geowissenschaften oder dem Ingenieurwesen angewandt wird. Die dritte Publikation stellt einen Diskussionsanstoß dar, IML als eine Form der Sensitivitätsanalyse zu betrachten und integriert somit Ansätze aus dem IML in eine größere Anzahl von Forschungsarbeiten zur Erklärung komplexer Systeme.

Die vierte Publikation verknüpft IML mit Interpretationen in der angewandten Statistik. Sie präsentiert einen modell-agnostischen Interpretationsansatz mit Hilfe von Forward Marginal Effects (FMEs), um jegliches Vorhersagemodell auf der lokalen, regionalen und globalen Ebene zu interpretieren: Der FME zeigt die Änderung der Vorhersage für spezifizierte Änderungen der Eingangsvariablen an; ein Non-Linearity Measure stellt zusätzliche diagnostische Information bereit, ob der FME ein ausreichender lokaler Deskriptor der Vorhersagefunktion ist; der Conditional Average Marginal Effect aggregiert lokale Modellerklärungen ohne die Treue zum zugrundeliegenden Modell zu verlieren. Zusätzlich stellt die fünfte Publikation das R-Paket `fmeffects` vor, welches die erste Softwareimplementierung der Theorie um FMEs darstellt.

Trotz der praktischen Relevanz Cluster zu erklären, fokussiert sich die IML-Forschung fast ausschließlich auf das überwachte Lernen. Die sechste Publikation verknüpft IML mit der Erklärung von Clustern. Sie stellt FACT (Feature Attributions for Clustering) vor, ein allgemeines Konzept von Arbeitsschritten (Sampling, Intervention, Reassignment, Aggregation) für algorithmus-agnostische Cluster-Interpretationsmethoden. Darüber hinaus werden zwei neuartige Interpretationsmethoden vorgestellt: SMART (Scoring Metric after Permutation) misst Änderungen in Cluster-Zuweisungen mit spezifizierbaren Scoring-Funktionen, nachdem ein Teil der Eingangsvariablen permutiert wurde; IDEA (Isolated Effect on Assignment) gibt lokale und globale Änderungen in Cluster-Zuweisungen wieder, die durch Änderungen an den Eingangsvariablen verursacht wurden.

Contents

Acronyms

AI artificial intelligence.

ALE accumulated local effects.

AME average marginal effect.

cAME conditional average marginal effect.

DBSCAN density-based spatial clustering of applications with noise.

DGP data generating process.

DME derivative ME.

FACT feature attributions for clustering.

FD finite difference.

FME forward marginal effect.

G2PC global permutation percent change.

GE generalization error.

HDMR high-dimensional model representation.

HPO hyperparameter optimization.

ICE individual conditional expectation.

IDEA isolated effect on assignment.

IML interpretable machine learning.

LIME local interpretable model-agnostic explanations.

LM linear model.

ME marginal effect.

ML machine learning.

NLM non-linearity measure.

PD partial dependence.

PFI permutation feature importance.

RF random forest.

SA sensitivity analysis.

SIPA sampling, intervention, prediction, aggregation.

SL supervised learning.

SMART scoring metric after permutation.

UL unsupervised learning.

XAI explainable artificial intelligence.

Part I.

Summary and Discussion

1 Overview

The term machine learning (ML) may be traced back to Samuel (1959) and could be described as providing computers with the ability to solve tasks from experience. This experience is encapsulated in mathematical models derived from data. The abundant availability of data and computational resources in the twenty-first century has propelled ML to the forefront of scientific and economic progress. Often used synonymously with the term artificial intelligence (AI), it is both a driver of automatization and knowledge discovery. Applications are as diverse as *medicine* (Rajkomar et al. 2019; Boulesteix et al. 2020), *psychology* (Dwyer et al. 2018), *economics* (Mullainathan et al. 2017; Athey et al. 2019), *finance* (Goodell et al. 2021), *text mining* (Žižka et al. 2021), *speech processing* (Deng et al. 2013), *robotics* (Pierson et al. 2017), *climate modeling* (Dueben et al. 2018), or *(video) games* (Skinner et al. 2019; Silver et al. 2016).

The process of interpreting models has become an integral aspect of ML. This young and quickly evolving field of research is typically referred to as interpretable machine learning (IML) (Molnar 2022) or explainable artificial intelligence (XAI) (Kamath et al. 2021). In recent years, there has been an upsurge of interest in IML with applications such as *neuroscience* (Fellous et al. 2019), *judicial settings* (Deeks 2019), *surgical decision support systems* (Gordon et al. 2019), *natural language processing* (Danilevsky et al. 2020), *drug discovery* (Jiménez-Luna et al. 2020), *digital pathology* (Evans et al. 2022), *robotics* (Das et al. 2021), *software engineering* (Tantithamthavorn et al. 2021), *cybersecurity* (Sharma et al. 2022), *education* (Khosravi et al. 2022), *earth observation* (Gevaert 2022), or *power systems* (Machlev et al. 2022).

However, the motivation to explain mathematical models is neither new nor limited to ML. For instance, in the earth sciences, models are typically explained via sensitivity analysis (SA) (Razavi et al. 2021). An innate connection between IML and SA exists, as both fields are based on similar principles and methodological approaches to explain mathematical models. The research community has not fully acknowledged this connection, resulting in potential research gaps and related work not being referenced sufficiently. Further inspiration can be drawn from applied statistics where interpretability plays a major role in modeling tasks. Comprehensible and actionable model explanations of the form *"how does the predicted outcome change if a set of features changes by specified amounts?"* are known as marginal effects (MEs) in many domain sciences such as *econometrics* (Greene 2019), *psychology* (McCabe et al. 2022), or *medical research* (Onukwugha et al. 2015) but are not

acknowledged in ML. Finally, many model-agnostic techniques are potentially applicable to the unsupervised clustering setting. In spite of the practical relevance of explaining clusters, IML largely focuses on supervised learning (SL). All this is an indication that IML could profit from bridging gaps to other fields and adopting a broader perspective on the interpretation of mathematical models.

1.1. Research Objectives

This dissertation is comprised of six papers contributing to the science of interpreting mathematical models, albeit in quite different aspects. It thus may be seen as a journey through different fields guided by the theme of interpretability. Specifically, I pursue three objectives in my research:

Sensitivity Analysis. Working towards a change of direction in the research community to synthesize ML and SA, better reference related work, and exploit research gaps.

Marginal Effects. Adapting MEs for the application to non-linear models, thereby establishing them as a valuable model-agnostic interpretation method.

Cluster Explanations. Adapting interpretation approaches from SL to the unsupervised clustering setting.

1.2. Outline

This dissertation consists of two major parts. In the first part, I will provide the reader with general background information on IML, other types of explanations connected to ML, and SA. Furthermore, I will summarize the key contributions of each paper and discuss how they relate to the overarching theme of bridging gaps in IML, potential avenues for development, and future developments of the field. The second part includes each contributing paper and the corresponding declaration of author contributions.

Sensitivity Analysis. We first venture out into the field of SA and gain new perspectives on existing interpretation methods. General principles seem to underlie the rapidly growing amount of techniques to explain models. The first bridge aims at discovering such principles and establishing a connection between methods both within IML and across different domains (where many interpretation methods are broadly compiled under the umbrella term of SA). Chapter 5 (Scholbeck et al. 2020) reveals that many model-agnostic interpretation methods are based on the same work stages, which is termed the SIPA (sampling, intervention, prediction, aggregation) framework. Model-agnostic methods let

us decouple the modeling process from the interpretation process but are subject to certain pitfalls, including the interpretation of models that do not generalize well, ignoring feature dependencies and interactions, or making unjustified causal interpretations. Chapter 6 (Molnar et al. 2022) provides guidance on identifying such common pitfalls and potential solutions. Many pitfalls stem from the SIPA methodology, which analyzes the sensitivity of the model output or model performance with respect to variations in feature values. Similar approaches are used to explain the hyperparameter optimization (HPO) process (Hutter et al. 2014; Moosbauer et al. 2021). Chapter 7 (Scholbeck et al. 2023b) provides an impetus for discussion on how IML can be seen as SA applied to ML, which integrates recent developments in IML into the larger body of research in SA on how to explain general complex systems.

Marginal Effects. The next step on our journey is applied statistics, which in the context of this thesis refers to domain sciences that utilize classic statistical models such as (generalized) linear or generalized additive models. This includes *econometrics* (Greene 2019), *psychology* (McCabe et al. 2022), or *medical research* (Onukwugha et al. 2015). In such fields, interpretability plays a major role in most modeling tasks, which strongly contrasts with the algorithmic black box nature of modeling in ML (Breiman 2001b). To interpret the above-mentioned white box model types, MEs (Williams 2012) (which refer to derivatives of the prediction function with respect to a feature or changes in prediction due to changes in feature values) are used to interpret the feature-target relationship. Although MEs are typically computed analytically, due to the model equation being known in such scenarios, they can be formulated in a model-agnostic way via forward differences, resulting in simple and valuable model-agnostic interpretations of the form *"how does the predicted outcome change if a set of features changes by specified amounts?"*. Chapter 8 (Scholbeck et al. 2024) bridges the gap to applied statistics by introducing a unified definition of forward marginal effects (FMEs) as a model-agnostic interpretation method with options to explain models on the local (observation-wise), regional (concerning subgroups), and global level (concerning the entire feature space). In addition, Chapter 9 (Löwe et al. 2023) describes the R package `fmeffects`, the first software implementation of the theory surrounding FMEs.

Cluster Explanations. The last step on our journey brings us to unsupervised learning (UL), or more specifically, clustering. While the recent wave of publications in IML has almost exclusively focused on SL (Molnar 2022), unsupervised methods such as clustering have been mostly ignored. In spite of the practical relevance, addressing the issue of cluster interpretability has been limited (Bertsimas et al. 2021). Chapter 10 (Scholbeck et al. 2023a) introduces a novel algorithm-agnostic approach to explain assignments of observations to clusters. This represents an evolution of thought from model-agnostic interpretations in SL to algorithm-agnostic interpretations in clustering. First, the SIPA framework from Chapter 5 is adapted to the unsupervised clustering setting where the prediction stage is replaced by a reassignment stage; based on this

conceptual approach for the clustering setting, two interpretation methods termed the scoring metric after permutation (SMART) and the isolated effect on assignment (IDEA) are developed. SMART measures changes in cluster assignments by custom scoring functions after permuting selected features. IDEA indicates local and global changes in cluster assignments after making uniform changes to selected features.

1.3. Overview of Contributing Papers

Chapter 5. Scholbeck, C. A., Molnar, C., Heumann, C., Bischl, B., and Casalicchio, G. (2020). "Sampling, Intervention, Prediction, Aggregation: A Generalized Framework for Model-Agnostic Interpretations". In: *Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part I*. Ed. by Cellier, P. and Driessens, K. Vol. 1167. Communications in Computer and Information Science. Cham: Springer International Publishing, pp. 205–216. DOI: 10.1007/978-3-030-43823-4_18

Chapter 6. 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: *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*. Ed. by Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.-R., and Samek, W. Vol. 13200. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 39–68. DOI: 10.1007/978-3-031-04083-2_4

Chapter 7. Scholbeck, C. A., Moosbauer, J., Casalicchio, G., Gupta, H., Bischl, B., and Heumann, C. (2023b). "Position Paper: Bridging the Gap Between Machine Learning and Sensitivity Analysis". In: arXiv: 2312.13234 [cs.LG]

Chapter 8. Scholbeck, C. A., Casalicchio, G., Molnar, C., Bischl, B., and Heumann, C. (2024). "Marginal Effects for Non-Linear Prediction Functions". In: *Data Mining and Knowledge Discovery* 38.5, pp. 2997–3042. DOI: 10.1007/s10618-023-00993-x

Chapter 9. Löwe, H., Scholbeck, C. A., Heumann, C., Bischl, B., and Casalicchio, G. (2023). "fmeffects: An R Package for Forward Marginal Effects". In: arXiv: 2310.02008 [cs.LG]

Chapter 10. Scholbeck, C. A., Funk, H., and Casalicchio, G. (2023a). "Algorithm-Agnostic Feature Attributions for Clustering". In: *Explainable Artificial Intelligence: First World Conference, xAI 2023, Lisbon, Portugal, July 26–28, 2023, Proceedings, Part I*. Ed. by Longo, L. Vol. 1901. Communications in Computer and Information Science. Cham: Springer Nature Switzerland, pp. 217–240. DOI: 10.1007/978-3-031-44064-9_13

2 Background

2.1. Notation and Preliminaries

The nature of the data dictates how the machine learns, predicts, and how its predictions can be explained. This dissertation is concerned with ML on structured, tabular data. SL (main subject of Chapters 5 to 9) and UL (main subject of Chapter 10) are two major learning paradigms.

2.1.1. Supervised Machine Learning

SL is concerned with learning a relationship between features and a target variable[1] based on labeled training data. We assume observations $\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)$ with $\boldsymbol{x}^{(i)} \in \mathcal{X}$ and $y^{(i)} \in \mathcal{Y}$ are drawn i.i.d. from an unknown data generating process (DGP) which is denoted by $\mathbb{P}_{\boldsymbol{X},Y}$, where $\boldsymbol{X} = (X_1, \ldots, X_p)$ denotes the random variables associated with the p-dimensional feature space $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p$, and Y denotes the random variable associated with the target space $\mathcal{Y}$. A subset of features is indexed by the set $S \subseteq \{1, \ldots, p\}$. The complement feature set is denoted by $-S = \{1, \ldots, p\} \setminus S$. The feature vector $\boldsymbol{x}^{(i)}$ can be partitioned into vectors of feature values $\boldsymbol{x}_S^{(i)}$ and $\boldsymbol{x}_{-S}^{(i)}$. With slight abuse of notation, we denote $\boldsymbol{x}^{(i)}$ by $\left(\boldsymbol{x}_S^{(i)}, \boldsymbol{x}_{-S}^{(i)}\right)$, regardless of the feature indices in S. For a single feature of interest, S is replaced by j. We have access to a data set $\mathcal{D} = \left((\boldsymbol{x}^{(i)}, y^{(i)})\right)_{i \in \{1,\ldots,n\}}$. A learning algorithm (also referred to as inducer) $\mathcal{I}$ learns the prediction function $\widehat{f} : \mathcal{X} \to \mathcal{Y}$ from a training subset $\mathcal{D}_{\text{train}} \subset \mathcal{D}$ with n_{train} observations, a process that is dictated by the hyperparameter configuration $\boldsymbol{\lambda} \in \Lambda$:

$$\mathcal{I} : (\mathcal{D}_{\text{train}}, \boldsymbol{\lambda}) \mapsto \widehat{f}$$

The space of learnable models is restricted to the hypothesis space $\mathcal{H}$:

$$\widehat{f} \in \mathcal{H}$$

[1] For regression tasks, we typically model a univariate target, while in classification, the dimensionality of the target depends on the number of classes. The explanation methods discussed in this thesis can easily be extended to multi-output prediction problems if they are applied to a single output at a time.

The model is trained such that it is able to generalize well given unseen test data. The model's generalization error (GE) with respect to the loss function L is defined as:

$$\text{GE} = \mathbb{E}_{(\boldsymbol{x},y)\sim\mathbb{P}_{\boldsymbol{X},Y}}\left[L\left(\widehat{f}(\boldsymbol{x}), y\right)\right]$$

As $\mathbb{P}_{\boldsymbol{X},Y}$ is generally unknown, most learners use empirical risk minimization to train the model:

$$R_{\text{emp}}(\widetilde{f}) = \frac{1}{n_{\text{train}}} \sum_{i\,:\,(\boldsymbol{x}^{(i)},\, y^{(i)})\,\in\,\mathcal{D}_{\text{train}}} L\left(\widetilde{f}(\boldsymbol{x}^{(i)}), y^{(i)}\right)$$
$$\widehat{f} = \underset{\widetilde{f}}{\text{argmin}}\, R_{\text{emp}}(\widetilde{f})$$

The final performance of the trained model is evaluated by a general performance measure ρ (which may coincide with L) on a test subset $\mathcal{D}_{\text{test}} \subset \mathcal{D}$:

$$\rho : \begin{cases} \mathcal{X} \times \mathcal{Y} \times \mathcal{H} \to \mathbb{R} \\ (\mathcal{D}_{\text{test}}, \widehat{f}) \mapsto \widehat{\text{GE}} \end{cases}$$

Resampling and aggregating results has been demonstrated to be an efficient use of data (Bischl et al. 2023); for instance, $\mathcal{D}$ can be split up into different training and test sets in an outer loop while $\mathcal{D}_{\text{train}}$ can be further split up into a training and test set to validate the model in an inner loop for HPO. In addition to controlling the behavior of the learner $\mathcal{I}$, we can configure the entire learning procedure via $\boldsymbol{\lambda}$, for example via the specification of resampling splits.

2.1.2. Clustering

UL is concerned with discovering patterns in unlabeled training data where no designated target variable is observed. A large part of UL concerns clustering (Tomašev et al. 2016) which aims at partitioning a data set of unlabeled data $\mathcal{D} = \left(\boldsymbol{x}^{(i)}\right)_{i\in\{1,\ldots,n\}}$ into k clusters $\mathcal{D}^{(c)} \subset \mathcal{D},\ c \in \{1,\ldots,k\}$ such that $\mathcal{D}^{(c_1)} \cap \mathcal{D}^{(c_2)} = \emptyset\ \forall\ c_1, c_2 \in \{1,\ldots,k\}\ \wedge\ c_1 \neq c_2$. The data are partitioned so that observations within clusters are more similar (or in closer proximity) to each other than to observations in other clusters. Through clustering, we can identify an underlying structure in the data, which can also be used for data compression (Jain 2010). As noted by Jain et al. (1988), it is crucial how to define and measure proximity, which is inherently context-dependent.

This resulted in thousands of clustering algorithms with a large spectrum of methodological approaches that can be categorized in different ways (Jain 2010; Xu et al. 2015; Ezugwu et al. 2022). Well-established categories include partitional methods, such as k-means

clustering (MacQueen 1967); hierarchical methods, such as finding nested partitions of the data with a top-down or bottom-up approach (Han et al. 2012); density-based methods, such as density-based spatial clustering of applications with noise (DBSCAN) (Ester et al. 1996); or grid-based methods such as statistical information grid-based clustering (Wang et al. 1997).

In contrast to SL, no ground truth label exists, and thus evaluating the quality of the clustering is more complex. Human experts may provide a proxy for a ground truth clustering, whose closeness to the found clustering can be evaluated with extrinsic methods such as the cluster homogeneity with respect to the ground truth label (Han et al. 2012). Given no such proxy, clusterings are evaluated with intrinsic methods which typically quantify the compactness and separation of clusters such as the silhouette coefficient (Rousseeuw 1987). Selecting an appropriate number of clusters k is paramount for a high-quality clustering. Some algorithms such as k-means clustering require the number of clusters k to be specified as a hyperparameter, while other algorithms such as DBSCAN adaptively search for the optimal number of clusters. In this regard, the elbow criterion (Thorndike 1953) is an established heuristic that takes into account the trade-off between too few or too many clusters.

2.2. Introduction to Interpretable Machine Learning

2.2.1. History

Molnar et al. (2020) provide an insight into the history of IML: Interpretable predictive models can be dated back as far as the early 1800s; but whereas the development of the support vector machine (Vapnik et al. 1974) or neural networks and deep learning (Schmidhuber 2015) revolutionized ML throughout the second half of the twentieth century, interpretations of such black box models only became a concern in the 2000s, e.g., with the introduction of a built-in feature importance measure for random forests (RFs) (Breiman 2001a); the mid 2010s marked an upsurge of interest in model interpretations, resulting in novel model-agnostic interpretation methods, such as local interpretable model-agnostic explanations (LIME) (Ribeiro et al. 2016), and novel interpretable model types, such as rule-based methods (Angelino et al. 2018). The 2010s could therefore be considered the defining time period for the crystallization of IML as a recognized field of research.

The term IML historically pertained to model explanations, but one could argue that any effort to interpret ML on some level shall be included under the same umbrella term. For instance, the emergence of novel HPO approaches such as Bayesian optimization (Jones et al. 1998; Hutter et al. 2011; Snoek et al. 2012) was accompanied by efforts to better understand the HPO process (Hutter et al. 2014; Moosbauer et al. 2021), and an increasing number of publications is exploring cluster explanations (De Koninck et al. 2017; Bertsimas

et al. 2021; Lawless et al. 2023). Throughout this dissertation, the reader may notice my efforts to expand the perspective on IML to include more types of explanations than mere model explanations.

2.2.2. Reflections on Interpretability

Interpretability and explainability are often used synonymously but can also be understood as different concepts (Marcinkevičs et al. 2023; Lipton 2018). For instance, Rudin (2019) refers to IML as the design of intrinsically interpretable models and to explainable ML as the application of post-hoc interpretation methods to an existing, typically black box, model. In the context of this dissertation, I do not differentiate between an interpretation and an explanation.

A pressing question surrounding IML is how to define interpretability which, unfortunately, has been insufficiently answered (Lipton 2018). Interpretability is often misused as a means to an end instead of an end in itself (Krishnan 2020). It is generally defined opportunistically to demonstrate a method's effectiveness, resulting in futile attempts to objectively compare and benchmark interpretation methods. Most discussions on interpretability pertain to models (Molnar 2022; Molnar et al. 2024a; Krishnan 2020; Lipton 2018; Rudin et al. 2022; Rudin 2019). However, the principles discussed below can be easily applied to explanations of other aspects of ML such as HPO or clustering.

Why Interpretability. Molnar et al. (2024a) formulate three goals of explanations, inspired by Adadi et al. (2018):

Explain to justify: By understanding models, their decisions can be justified, particularly if results are unexpected. A better understanding of models establishes trust in their decision making capabilities and may even be a prerequisite to employ them in regulatory environments (Wachter et al. 2018; Lipton 2018).

Explain to improve: Especially in high-stakes decision making contexts, knowing when models fail is paramount. We therefore need to understand the processes dictating how they operate to counteract and prevent malfunctionings (Krishnan 2020). Malfunctionings can manifest in different ways. For one, they depend on formal requirements set out in advance. For instance, regulations can be put in place to prevent a model learning to discriminate against minorities (Goodman et al. 2017), which links back to the goal of *explaining to justify*. Furthermore, understanding what the model has learned lets us improve its predictive performance.

Explain to discover: Models are now able to find patterns in data sets of unprecedented magnitudes. Coupled with model explanations, these patterns can be uncovered and expressed in a comprehensible way.

Attempts at Defining Interpretability. Interpretability may be defined as the degree to which humans are able to understand the model's decision making process (Miller 2019). A similar definition measures interpretability as the degree to which a human can predict the model's decision (Kim et al. 2016). Although human comprehension inarguably plays an important role in interpretability, it is neither objectively measurable for individuals (multiple models rated by a single individual), nor across multiple individuals (a single model rated by multiple individuals), thus lacking necessary mathematical precision. Interpretability is closely connected to complexity and sparsity (Rudin et al. 2022; Molnar et al. 2020). Although better quantifiable, this shifts the burden to (also quite arbitrary) complexity and sparsity measurements which, furthermore, are not a sufficient descriptor of interpretability. For instance, a regression model with a single cubic term is more sparse than a model with two linear terms but less comprehensible to the human mind. In other domains, interpretability is connected to the infusion of prior knowledge, e.g., about physical processes, into the modeling process (Zhang et al. 2020). It is now widely agreed upon that interpretability is a multi-faceted concept and could include complexity / sparsity, the ability for disentanglement of information flows, generative constraints such as in physics, manual control of the learning process, the ability for visual interpretations, and more (Rudin et al. 2022). Scientific explanations are also widely discussed in philosophy of science. By differentiating between explanandum (what is explained) and explanans (how it is explained), explanations can be approached via logical models such as the deductive-nomological model (Hempel et al. 1948). Due to the many different philosophical models for scientific explanations (Woodward et al. 2021), philosophical approaches are not further discussed in the context of this dissertation. The fact that we cannot concisely define interpretability does not, however, diminish the scientific contributions of interpretation methods (Krishnan 2020).

A Case for Goal-Oriented Interpretations. The missing conceptual foundation of IML carries the risk of techniques being applied without a clear vision of how the model shall be interpreted. A proper strategy should consist of defining the goals (or purpose) of the interpretation upfront, then deciding on an appropriate method (Freiesleben et al. 2023). Chen et al. (2022) argue to treat IML methods as diagnostics for ML, similar to classic statistical diagnostics such as error bars or hypothesis tests, with clear guidelines for when and how to apply a method. They use the analogy of a doctor assessing a patient's overall health with diagnostic tools such as x-rays or blood pressure measurements; neither tool is a sufficient descriptor of the patient's overall health but can provide crucial information on specific questions, e.g., whether the patient suffers from a bone fracture or of cardiovascular disease. In other words, the diagnostic tool is linked to a specific use case. Freiesleben et al. (2024) enlarge upon this idea and advocate a step-wise approach for scientific inference with IML: first, formalize a scientific question and establish whether IML is able to answer it, then design a model property descriptor (in other words, an interpretation technique) to answer the question and estimate it, and furthermore, quantify the uncertainty connected with the model property description.

2.2.3. Principles of Model Interpretations

Model Structure. Interpretable models provide built-in measures informing about the relationship between features and the predicted outcome. Regression models with known model equations (such as (generalized) linear or generalized additive models), decision trees, or k-nearest-neighbors are well-established interpretable model types.

Consider the model equation of a linear model (LM) for an observation $\boldsymbol{x}$:

$$\widehat{f}(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + \ldots \beta_p x_p + \epsilon \tag{2.1}$$

where ϵ denotes the residual. A coefficient such as β_1 provides immediate insight into the direction and magnitude of the change in predicted outcome due to a change in the value of x_1. Such a model description can be referred to as a feature effect (Scholbeck et al. 2020; Casalicchio et al. 2019).

We can also interpret β_1 as the importance of the associated feature, as a large effect on the predicted outcome makes a feature more important to the model behavior. Feature importance is also linked to uncertainty; for instance, if x_1 is associated with a large effect on the prediction, but the uncertainty associated with x_1 is low, we could argue that it is not an important feature of the model. Alternatively, the importance of a feature can be measured by evaluating its impact on the model performance (Casalicchio et al. 2019).

The influence of features on the predicted outcome can also be referred to as a feature attribution (Zhou et al. 2022). Unfortunately, a standardized terminology for IML does not yet exist.

Given its comprehensible model equation, the LM could be considered the prime example for an interpretable model. But even with access to a model equation, the workings of the model might be difficult to interpret. By introducing non-linear feature terms or interactions, we quickly lose comprehension of the individual effects of features. Recall that interpretability can be considered a multi-faceted concept, where sparsity or complexity of the model plays a major role.

Scope of Explanation. We need to further differentiate between local, regional, and global explanations: Local explanations explain the model for a single data point; regional explanations explain the model for a region of the feature space; and global explanations explain the model for the entire feature space. Given randomly sampled data, global explanations can be estimated by considering an entire data set, whereas regional explanations can be estimated by considering subsamples from specific subspaces. The contributing paper by Scholbeck et al. (2020) in Chapter 5 showcases how in general, global model explanations result from aggregating local explanations. For the LM, local, regional, and global explanations coincide, but they can differ considerably for non-linear models. Therefore, it is important to carefully select the appropriate explanation scope for the task at hand.

2.2.4. Classic Model-Agnostic Methods

In recent years, model-agnostic techniques have become the centerpiece of IML (Molnar et al. 2020). They can be used to generate insight into the influence of features on the predicted outcome for any predictive model. But they are also applicable to interpretable models, where they may provide additional, crucial insight. For instance, a decision tree with binary splits is highly interpretable, due to decision rules resembling the human thought process, but it does not provide a metric indicating a feature's importance, which model-agnostic methods can provide. The contributing paper by Scholbeck et al. (2020) in Chapter 5 demonstrates that model-agnostic methods work by querying the model with different feature values, a methodology akin to SA. This connection is further explored in the contributing paper by Scholbeck et al. (2023b) in Chapter 7.

Popular local interpretation methods include the individual conditional expectation (ICE) (Goldstein et al. 2015), LIME (Ribeiro et al. 2016), anchors (Ribeiro et al. 2018), counterfactual explanations (Wachter et al. 2018), Shapley values (Štrumbelj et al. 2010), or Shapley additive explanations (Lundberg et al. 2017).

Popular global methods include the partial dependence (PD) (Friedman 2001), Shapley additive global importance (Covert et al. 2020), the permutation feature importance (PFI) (Fisher et al. 2019), or accumulated local effects (ALE) (Apley et al. 2020).

As the list of available methods is vast and growing steadily, this section will only illustrate methods that are particularly relevant for the contributing papers in this dissertation, namely the ICE, PD, PFI, and LIME. For a more comprehensive overview, the interested reader may be referred to the work of Molnar (2022).

Individual Conditional Expectation and Partial Dependence. The ICE represents the prediction of an SL model $\widehat{f}$ for a single observation $\boldsymbol{x}^{(i)}$ while replacing $\boldsymbol{x}_S^{(i)}$ by $\tilde{\boldsymbol{x}}_S$ and keeping the observed values $\boldsymbol{x}_{-S}^{(i)}$ fixed:

$$\text{ICE}_{\boldsymbol{x}^{(i)},S}(\tilde{\boldsymbol{x}}_S) = \widehat{f}(\tilde{\boldsymbol{x}}_S, \boldsymbol{x}_{-S}^{(i)})$$

The PD represents the global, average influence of a set of features. It can be formulated for the DGP or as a Monte-Carlo estimate (which corresponds to an aggregation of ICEs):

$$\text{PD}_S(\tilde{\boldsymbol{x}}_S) = \int \widehat{f}\,(\tilde{\boldsymbol{x}}_S, \boldsymbol{X}_{-S})\; d\,\mathbb{P}_{\boldsymbol{X}_{-S}}$$

$$\widehat{\text{PD}}_{\mathcal{D},S}(\tilde{\boldsymbol{x}}_S) = \frac{1}{n}\sum_{i=1}^{n} \widehat{f}\left(\tilde{\boldsymbol{x}}_S, \boldsymbol{x}_{-S}^{(i)}\right)$$

Computing ICEs and averaging out the influence of the remaining features requires the generation of artificial combinations of feature values. If $\boldsymbol{X}_S$ and $\boldsymbol{X}_{-S}$ are dependent, such

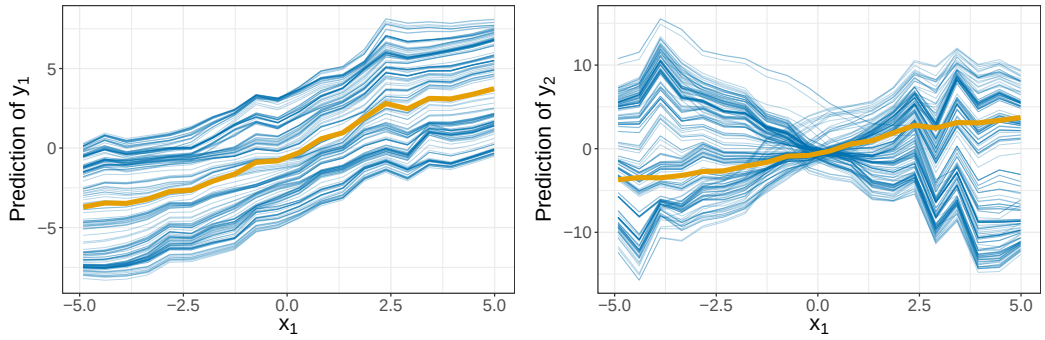


Figure 2.1.: ICEs and PD for two different models. **Left:** Similarly-shaped ICE curves with vertical differences indicate an additive influence of other features. **Right:** Differently-shaped ICE curves indicate an interaction between x_1 and other features.

artificial instances might be unlikely to occur in reality, thus creating a biased estimate of the feature effect. Ignoring feature dependencies is a common pitfall of model-agnostic interpretations, which will be discussed in the contributing paper by Molnar et al. (2022) in Chapter 6.

The ICE and PD are prominent tools due to their diagnostic capabilities and simplicity. In univariate ICE plots, vertical differences between ICEs inform us about an additive influence of other features on the prediction, while different trajectories of the ICE curves tell us about an interaction.

As an illustration, consider the following DGP:

$$x_1 \sim \text{Unif}(-5, 5) \qquad x_2 \sim \text{Unif}(-5, 5) \qquad \epsilon \sim N(0, 1)$$
$$y_1 = x_1 + x_2 + \epsilon \qquad y_2 = x_1 x_2 + \epsilon$$

We train two RFs to predict y_1 and y_2 on 200 generated samples. Fig. 2.1 visualizes ICEs and the PD for the influence of x_1 on the predicted y_1 (left) or y_2 value (right). For both DGPs, the ICE and PD plot provides model diagnostics about the isolated effect of x_1 on the local and global level and whether there are interactions with other features.

Permutation Feature Importance. Another option is to evaluate how querying the model with different feature values affects the model performance. The PFI is based on a simple but compelling principle: if shuffling feature values—which destroys the mutual information between a feature and the target—results in a loss in performance, this feature must have been important.

It is generally advisable to use test data to compute the PFI (Molnar 2022). To be precise, assume we shuffle the j-th column in a test set $\mathcal{D}_{\text{test}}$ and denote the shuffled data set by $\tilde{\mathcal{D}}_{\text{test},j}$. The PFI is estimated as:

$$\widehat{\text{PFI}}_j = \rho(\tilde{\mathcal{D}}_{\text{test},j}, \widehat{f}) - \rho(\mathcal{D}_{\text{test}}, \widehat{f})$$

This principle has since been extended to groups of features (Au et al. 2022). However, shuffling also destroys the mutual information between the feature and other features, thus distorting the influence of interactions. A potential remedy is to shuffle conditionally on the values of other features (Molnar et al. 2024b).

Local Interpretable Model-Agnostic Explanations. LIME is a local technique that approximates the black box model in the vicinity of a single data point with an interpretable surrogate model. LIME predicts with (artificially) sampled instances and uses a kernel function to weight the predictions by the instances' proximity to the point of interest. Afterwards, an interpretable surrogate model is trained to predict the weighted predictions. Consider a surrogate model g, a function $\pi_{\boldsymbol{x}}$ where $\pi_{\boldsymbol{x}}(\boldsymbol{z})$ indicates the proximity of an instance $\boldsymbol{z}$ to the instance of interest $\boldsymbol{x}$, a complexity measure C for the surrogate model, and a measure U indicating the discrepancy between the surrogate model g and the black box model $\widehat{f}$ in the locality defined by $\pi_{\boldsymbol{x}}$. Formally, the surrogate model is found by minimizing U with respect to g while limiting the complexity of g:

$$\underset{g}{\text{argmin}} \quad U(\widehat{f}, g, \pi_{\boldsymbol{x}}) + C(g)$$

2.3. Other Types of Explanations

2.3.1. Marginal Effects

MEs have been historically associated with applied statistics (Williams 2012; Arel-Bundock 2023) and are not included in the classic toolbox of model-agnostic IML methods (Molnar 2022). MEs are often motivated by the desire to achieve an interpretation similar to a beta coefficient for generalized linear models (McCabe et al. 2022). We first need to differentiate between MEs for continuous and categorical features.

Changes in Continuous Features. Predominantly in econometrics, MEs for continuous features are defined in terms of derivatives of the prediction function with respect to a feature (Greene 2019), which I will refer to as a derivative ME (DME):

$$\text{DME}_{\boldsymbol{x},j} = \frac{\partial \widehat{f}(\boldsymbol{x})}{\partial x_j}$$

The derivative can be numerically approximated with finite differences (FDs). A popular choice is a forward difference with a very small value of the step size h:

$$\text{DME}_{\boldsymbol{x},j} \approx \frac{\widehat{f}(x_j + h, \boldsymbol{x}_{-j})}{h} \quad , h > 0$$

The quality of the approximation generally increases for smaller step sizes but may deteriorate due to numerical factors such as cancellation errors (Sauer 2011).

An alternative, lesser known definition is based on forward differences, hereafter referred to as FME. We can easily define it for multiple feature changes:

$$\text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} = \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S})$$

Changes in Categorical Features. Categorical MEs correspond to the change in prediction when switching a reference category to another category (Williams 2012). For m categories, we receive $m - 1$ categorical MEs. Let the categories of the j-th feature be denoted by $C = \{c_1, \ldots, c_m\}$. We first select a reference category $c_r \in C$. For an observation $\boldsymbol{x}$, the categorical MEs correspond to:

$$(\text{ME}_{\boldsymbol{x},j,c_r,c_l})_{c_l \in C \setminus \{c_r\}} = \left(\widehat{f}(c_l, \boldsymbol{x}_{-j}) - \widehat{f}(c_r, \boldsymbol{x}_{-j})\right)_{c_l \in C \setminus \{c_r\}}$$

Problems of MEs. Although MEs can be applied to any predictive model, there are several obstacles: First, multiple definitions of MEs exist, and the definition of categorical MEs does not serve any goal-oriented interpretation; second, albeit its prominence in econometrics (Greene 2019), the DME is not particularly suited for effect interpretations, as a change in feature values results in a different prediction than implied by the derivative; third, the better-suited FME suffers from a loss in information about the shape of the prediction function along the forward difference; fourth, aggregating MEs to the global average marginal effect (AME), as it is typically done in many domain sciences (Onukwugha et al. 2015), is not sensible for highly non-linear models.

These points form the motivation for the contributing paper by Scholbeck et al. (2024) in Chapter 8, which introduces a unified definition of FMEs for continuous and categorical features, a non-linearity measure (NLM) to provide additional diagnostic information on whether the FME is a sufficient local descriptor of the prediction function, and the conditional average marginal effect (cAME) to estimate regional instead of global effects.

2.3.2. Cluster Explanations

In many applications, interpretability of the clustering result is a specific requirement (Han et al. 2012). However, Bertsimas et al. (2021) note that most clustering algorithms are not designed with interpretability (in the original feature space) in mind.

As a remedy, one could apply post-processing methods to the clustering result such as computing cluster prototypes, which fails for elongated or non-isotropic clusters; additional metrics such as the variance in each dimension of the cluster, which complicates interpretations by increasing the number of summary statistics; or dimensionality reduction and subsequent visualization of the clustering in two dimensions, which obfuscates the relationship between the original feature space and the clustering (Bertsimas et al. 2021).

Interpretable Clustering. A simple solution is to select an interpretable clustering algorithm, which provides intelligible information about the characteristics of a cluster. An interpretable algorithm may also search for more interpretable clusters (regarding characteristics defined by the algorithm). One option is to search for a clustering that can be expressed as a decision tree (Bertsimas et al. 2021; Fraiman et al. 2013). Interpretable clustering of numerical and categorical objects (Plant et al. 2011) is an information-theoretic approach that minimizes the minimum description length of each cluster and finds a rule description by assuming that clusters are multivariate normally distributed. Lawless et al. (2022) simultaneously search for clusters and construct polytopes around them that act as cluster descriptors.

Algorithm-Agnostic Cluster Explanations. In many applications, restricting the choice of clustering algorithm to interpretable ones is not an option. Similarly to model-agnostic methods for SL, algorithm-agnostic methods create insight into the clustering for any clustering algorithm. Such interpretations are typically post-hoc and consider a given, fixed clustering. A simple option is to train an SL classifier to predict the found cluster labels, which is interpreted instead (De Koninck et al. 2017). Carrizosa et al. (2022) select a representative instance within each cluster referred to as a prototype. Lawless et al. (2023) describe clusters by constructing polyhedrons around them.

Another option is to reassign instances to existing clusters after manipulating feature values, which essentially corresponds to an SA of the clustering result. The global permutation percent change (G2PC) indicates the fraction of observations being reassigned to different clusters after shuffling feature values; the local permutation percent change indicates the reassignment of a single instance after perturbing feature values (Ellis et al. 2021). For a given clustering $\left(\mathcal{D}^{(c)}\right)_{c\in\{1,\ldots,k\}}$, the post-hoc assignment of an observation $\boldsymbol{x}$ to a cluster index $c \in \{1, \ldots, k\}$ based on a pre-defined criterion can be formalized by the function a:

$$a : \begin{cases} \mathcal{X} \to \{1, \ldots, k\} \\ \boldsymbol{x} \mapsto c \end{cases}$$

2.3.3. HPO Explanations

Training Complexity and HPO. Understanding the HPO process is in large part dictated by the complexity of the training process. For instance, training LMs simply requires solving normal equations, a rather comprehensible process that is sufficiently understood by humans. For more complex model types such as RFs, the training process involves training many different regression trees, a process that is understood by humans conceptually but is too complex to comprehend in detail. This trickles down to the interpretability of the HPO process, which we typically are interested in: the less interpretable the training process, the less interpretable the HPO process.

Explanations of HPO. An increasing number of techniques provide insight into the relationship between $\boldsymbol{\lambda}$ and $\widehat{\text{GE}}$, which can be formalized by the function ψ:

$$\psi : \begin{cases} \Lambda \to \mathbb{R} \\ \boldsymbol{\lambda} \mapsto \widehat{\text{GE}} \end{cases}$$

Hutter et al. (2014) conduct a functional analysis of variance of the HPO process for an RF surrogate. Moosbauer et al. (2021) develop a PD for HPO with uncertainty estimate enhancements by exploiting uncertainty quantification mechanisms in Bayesian optimization. An ablation analysis (Fawcett et al. 2016; Biedenkapp et al. 2017) can be used to iteratively modify hyperparameter settings one parameter at a time and evaluate changes in the model performance.

2.4. Introduction to Sensitivity Analysis

2.4.1. Overview

SA is an auxiliary discipline used to explain complex mathematical systems with applications as diverse as *environmental modeling* (Song et al. 2015; Wagener et al. 2019; Shin et al. 2013; Haghnegahdar et al. 2017; Gao et al. 2023; Mai et al. 2022; Nossent et al. 2011), *engineering* (Guo et al. 2016; Ballester-Ripoll et al. 2019; Becker et al. 2011), *nuclear safety* (Saltelli et al. 2002), *biology* (Sumner et al. 2012), *energy management* (Tian 2013), *economics* (Harenberg et al. 2019; Ratto 2008), or *financial risk management* (Baur et al. 2004).

History. Razavi et al. (2021) reveal several facts about the historical origins and evolution of SA: While the basic notion of changing one or multiple factors to evaluate their effects on the outcome or a quantity of interest has a long history in science, the modern era of SA started to materialize in the 1970s and 1980s with the widespread adoption of computational

modeling and the extension of design of experiments to design of computer experiments (Santner et al. 2018); however, contributions to SA are scattered among different fields, resulting in a lack of visibility.

Systems Modeling. Saltelli et al. (2008) define SA as the study of how uncertainty in model output can be attributed to uncertainty in model inputs. A system consists of interconnected models (essentially functions) that can be data-driven or mechanistic (also referred to as process-based) (Razavi et al. 2021). An SA model, which can be any functional relationship between inputs and outputs, is therefore a more general notion than an ML model. Data-driven and mechanistic models are increasingly combined to form hybrid systems (Razavi 2021; Reichstein et al. 2019), but the principles and methods to explain systems are universal. To be precise, a system consists of multiple models ϕ mapping a vector of inputs $\boldsymbol{z} \in \mathcal{Z}$ to a vector of outputs $\boldsymbol{q} \in \mathcal{Q}$:

$$\phi : \begin{cases} \mathcal{Z} \to \mathcal{Q} \\ z \mapsto \boldsymbol{q} \end{cases}$$

Models within a system are typically interconnected such that outputs of one model are inputs to another model. This results in "trickle-down" effects where changing a system input has effects on multiple models. The time-dependent nature of many systems, e.g., in the earth sciences (Gupta et al. 2018), further complicates SA. For instance, the conversion of rainfall to runoff in hydrological systems can be subject to a certain delay (Beven 2012).

As an example, consider the earth system modeling framework (Collins et al. 2005; Hill et al. 2004), a software suite developed for the implementation of multi-component systems in the earth sciences. A component represents a physical domain such as an atmospheric, oceanic, or land surface model. Such component models may be independently developed and must be coupled together through appropriate software interfaces and data conversion protocols.

Such a system may now be analyzed for various purposes: assessing the accuracy of the modeled system in describing real world phenomena, identifying influential input factors, identifying regions in the input factor space that are most influential in determining output, or identifying interactions between input factors (Razavi et al. 2015).

How SA Relates to IML. Whereas IML started to materialize as a separate field in the 2010s, the origins of SA date back many decades. Both fields have similar motivations, but SA is a much broader field to explain any system of functional relationships (see Table 2.1). It can therefore be used as an overarching framework to interpret various aspects of ML, both conceptually and methodologically. This key message is expounded in the contributing article by Scholbeck et al. (2023b) in Chapter 7.

	IML	SA
Historical Origins	2000s and 2010s	1970s and 1980s
Scientific Venues	ML-centric	scattered among fields (e.g., *operations research*, *earth sciences*, *engineering*, etc.)
Model Types	data-driven	data-driven process-based
Explanation	feature-prediction feature-performance hyperparameter-performance	any system of interconnected models

Table 2.1.: A comparison of IML and SA in terms of their historical origins, what scientific venues papers are published in, the types of models that are explained, and the types of explanations.

2.4.2. Methods

We can characterize sensitivities within a system with a diverse spectrum of methods that can be categorized in multiple ways. Inspired by Razavi et al. (2021), I categorize methods as FD-based, distribution-based, or regression-based. Furthermore, many approaches use additional metamodels to reduce computational costs. Due to the vast number of publications on SA, this section only provides a short overview, raising no claim to completeness. For a more comprehensive overview (and slightly different categorizations), the interested reader may be referred to the works of Saltelli et al. (2008), Iooss et al. (2015), Borgonovo et al. (2016), or Razavi et al. (2021).

Finite-Difference-Based. These methods gather FDs in model output from multiple points of the input space. This includes (numeric) derivatives and larger step sizes, e.g., in the form of elementary effects. The elementary effect is a forward difference with large, varying step sizes and was first introduced as part of the Morris method (Morris 1991; Campolongo et al. 2007; Saltelli et al. 2008). The Morris method is an important representative of one-factor-at-a-time methods, which create paths through the input space by varying one factor at a time while keeping all remaining factors fixed. FD-based methods are often used for screening purposes due to their simplicity and low computational cost but are criticized for leaving important areas of the input space unexplored (Saltelli et al. 2010). Novel FD-based methods include derivative-based global sensitivity measures (Sobol et al. 2010), which average derivatives of the model with respect to inputs at locations obtained via random or

quasi-random sampling, and variogram analysis of response surfaces (Razavi et al. 2016), which summarizes the variance of FDs with equal distance across the input space.

Distribution-Based. This group of methods characterizes changes in the model output distribution resulting from variations in inputs, for instance by focusing on statistical moments. The most established subgroup of distribution-based methods is variance-based SA (Saltelli et al. 2008), which aims at attributing the variance in model output to the variance in model inputs. To capture various orders of interactions between features, the model is first decomposed into a high-dimensional model representation (HDMR) (Rabitz et al. 1999). The Sobol index (Sobol 1990) represents the fraction of output variance explained by individual terms in the HDMR. Other methods evaluate changes in the mean, skewness, or the curtosis of the output distribution (Dell'Oca et al. 2017). Focusing on particular moments is criticized for not fully characterizing the output distribution, which is what moment-independent or density-based methods aim to achieve (Borgonovo et al. 2016). Note that some authors refer to moment-independent methods as distribution-based SA to differentiate it from methods focusing on statistical moments such as variance-based SA (Borgonovo et al. 2012).

Regression-Based. This group of methods is connected to certain types of data-driven models that provide built-in measures to quantify the sensitivity of model output with respect to model inputs. For instance, recall the LM from Eq. (2.1) where the beta coefficient informs us about the feature influence on the predicted outcome. A modern, model-agnostic option is to leverage the additional diagnostic value of evaluating changes in model performance in data-driven modeling. As noted by Razavi et al. (2021), this includes approaches of including features one at a time and evaluating improvements in model fit, which is done by multivariate adaptive regression splines (Friedman 1991) for instance, or first training a model with all features and evaluating changes in performance when excluding them.

Metamodeling. A major component of SA concerns metamodeling, which refers to building surrogate models that approximate the original model but are cheaper to evaluate. Especially for distribution-based SA, the computation of an HDMR and the characterization of the output distribution requires many model evaluations. This is exacerbated by expensive model evaluations in many applications such as large-scale earth system models (Naz et al. 2023). Gaussian processes (Le Gratiet et al. 2017; Marrel et al. 2008; Marrel et al. 2009) and polynomial chaos expansion (Le Gratiet et al. 2017; Sudret 2008) are established metamodeling approaches. Metamodeling requires accounting for additional uncertainty associated with the metamodel (Razavi et al. 2021).

3 Discussion of Contributions

In this chapter, I will discuss the key contributions of each paper and how they relate to the overarching theme of bridging gaps in IML. Furthermore, I will point towards potential research gaps the corresponding paper did not manage to address.

Chapter 5. Scholbeck et al. (2020) propose a general framework of work stages for model-agnostic interpretation methods. The recent wave of research in IML (Molnar et al. 2020) created many new methods (such as ICEs, LIME, or the PFI) while older ones reentered the limelight (such as the PD). The paper demonstrates that this seemingly loosely connected set of methods is in fact based on the same methodological approach where changes in feature values are followed by predictions and various aggregations. This consolidates the literature by providing a unified view and terminology for model-agnostic interpretations.

The SIPA framework is a first step towards a general theory of model-agnostic interpretations but leaves several avenues for development unexplored, including the formation of a connection with SA and the definition of a unified mathematical framework. These points are partly revisited in the follow-up work in Chapter 7.

Chapter 6. Molnar et al. (2022) discuss several general pitfalls of model-agnostic interpretation methods such as ignoring feature dependencies, interactions and issues in high-dimensional settings, or making unjustified causal interpretations. Model-agnostic techniques are easily applied to a predictive model but require a careful prior assessment of the modeling task to avoid erroneous model interpretations. This work extends the perspectives gained in Chapter 5. We are now aware of a general methodology behind model-agnostic methods: the SIPA framework. This methodology, in connection with data-driven models, is the root cause of many pitfalls described in this chapter. Let me emphasize this by directly quoting the paper: *"Since many of the interpretation methods work by similar principles of manipulating data and 'probing' the model, they also share many pitfalls."* (Molnar et al. 2022).

The work in Chapter 5 and 6 is a thorough evaluation of model-agnostic methods but still lacks a comprehensive mathematical framework and axiomatic base. For instance, the axiom of sensitivity (Janzing et al. 2020; Sundararajan et al. 2017) dictates that if a model is not dependent on an input, the input's attribution shall be zero. It is conceivable that a comprehensible mathematical framework for IML could be (at least partially) axiomatic,

thus also putting the notion of pitfalls in certain scenarios on a more solid theoretical foundation.

Chapter 7. Scholbeck et al. (2023b) form a connection between IML and SA. This paper represents an evolution of thought from Chapter 5; whereas the SIPA methodology was demonstrated for many model-agnostic methods by Scholbeck et al. (2020), there was an evident but unexplored resemblance with SA, and interpretations of other ML processes such as HPO were still not taken into account. This position paper argues that IML can be seen as a form of SA, which integrates recent advances in IML into the larger body of research on how to interpret complex systems. It formally describes how ML can be viewed as a system suitable for SA and discusses how existing interpretation methods relate to this perspective. The goal of this paper is to work towards a better recognition of related work and the exploitation of potential research gaps that have arisen due to the concurrent development of similar interpretation methods in different communities. This paper further contributes to efforts of creating a unified theory of interpretations in ML. The scope is much broader than for the SIPA framework and includes various ML processes and model-specific interpretations.

On a less positive note, the paper factors out certain aspects of ML from the modeled system such as causal inference or UL, albeit potential connections are pointed out. Although it includes a more formal description of what functions interpretations in ML may operate on, it does not provide an exhaustive mathematical framework, thus only building a foundation for future work on the formal description of IML.

Chapter 8. This paper marks the second bridge, between IML and interpretations in applied statistics. Scholbeck et al. (2024) present a new theory of model-agnostic FMEs to interpret models on the local, regional, and global level. The FME is motivated by the notion of comprehensible and goal-oriented model explanations. Common research questions such as the effect of increasing a patient's age on the disease risk can be answered with FDs. They are easily computed, understood, and communicated to stakeholders. Although such interpretations are common in many domain sciences, they were not yet properly discussed in IML. I assume this partly stems from the incomprehensible literature on MEs and widespread confusion regarding their definition; Arel-Bundock (2023) describes this in his book aptly named *The Marginal Effects Zoo* accompanying the R package `marginaleffects`. The work by Scholbeck et al. (2024) is the first formal introduction of MEs in an ML context. It seizes the opportunity to introduce a unified, goal-oriented definition of FMEs for continuous and categorical features and to add multiple add-ons for non-linear prediction functions, namely the NLM and the cAME. The NLM is an additional diagnostic measure, indicating the degree to which the FME is a sufficient descriptor of the prediction function for a specific location in the feature space. The cAME combats an "aggregation bias" resulting from averaging heterogeneous local model explanations.

Another reason that MEs had not gained traction in the ML community may be that existing methods could be adapted to produce similar model explanations. For instance, the FME is equal to the difference between two values on an ICE, which is formally demonstrated in the paper. The FME itself therefore does not represent a completely novel concept in IML but rather *addresses, labels, and formalizes* an important interpretation use case that can be answered with forward differences by taking inspiration from applied statistics. Note that this criticism does not apply to the NLM and cAME, which are novel concepts that add to the diagnostic capabilities of FMEs in a meaningful way. However, there are potential avenues for development, including different implementations of the subgroup approach with cAMEs: while the paper chooses recursive partitioning to find subgroups with homogeneous FMEs, observations could, for instance, also be clustered as long as the resulting clusters can be explained intelligibly. The explanation of clusters is explored in Chapter 10.

Chapter 9. Löwe et al. (2023) describe the software package `fmeffects`, an R implementation of the theory presented in Chapter 8. This paper presents the first software implementation of the theory surrounding FMEs, including the NLM and the cAME. It is based on a modular software design to allow for future extensions and to facilitate maintainability.

I shall also discuss how `fmeffects` relates to the existing software ecosystem of MEs (in R). The R package `marginaleffects` (Arel-Bundock 2023) is the only viable alternative to compute FMEs: it is a comprehensive framework for various kinds of FD-based operations (including derivatives) on model objects in R and—in succession to the release of `fmeffects`—was extended with support of `mlr3` (Lang et al. 2019) model objects, which greatly enhances the user experience by supporting a large spectrum of ML models. However, `marginaleffects` does not completely cover the capabilities of `fmeffects`; specifically, it does not support the NLM and the cAME estimate via recursive partitioning. Furthermore, let me reinforce a key argument from Chapter 8 here: FMEs can resolve the ambiguity and confusion surrounding MEs with a unified definition and desirable goal-oriented interpretation, thereby making other definitions obsolete. The `fmeffects` package is designed accordingly and solely focuses on FMEs.

Chapter 10. This chapter bridges the third gap, between IML and cluster explanations. Scholbeck et al. (2023a) propose a general framework to design algorithm-agnostic cluster explanation methods termed feature attributions for clustering (FACT). The FACT framework consists of a sampling, intervention, reassignment, and aggregation stage. Furthermore, the paper presents the cluster explanation methods SMART and IDEA. Recall that in Chapter 5, the SIPA framework for model-agnostic methods was presented by Scholbeck et al. (2020), which serves as inspiration for FACT. This paper therefore forms a connection with earlier research on model-agnostic methods and SA and transfers various interpretation concepts to the unsupervised clustering setting. For SMART,

we first shuffle feature values (as for the PFI), compute a confusion matrix of cluster assignments before and after shuffling, and then summarize information contained in the confusion matrix via custom scoring functions. For IDEA, we make isolated changes to feature values (as for the ICE and PD), then visualize changes in cluster assignments on the local, regional, and global level. This paper represents one of the first works on algorithm-agnostic cluster explanations by the means of SA, hopefully creating an impetus for future research in this direction.

However, the concept of conducting an SA of cluster reassignments is not yet evaluated well enough to fully evaluate its practical effectiveness in explaining clusters. The field also does not provide cogent definitions of what an explanation of a cluster is, evidently a pervasive problem in all of interpretability research. Recall from Section 2.2.2 that the term *interpretability* is often used as a means to an end instead of an end in itself (Krishnan 2020). In the paper, we introduce our methodology as describing the importance or effects of features for assigning observations to existing clusters, thus making it subject to the exact same criticism. The fact that there typically is no ground truth clustering unless created by human experts (Han et al. 2012) (which, besides, is subject to human judgement) further exacerbates the issue of putting the field on a solid terminological foundation. Future work may also explore the assignment of new observations from a hold-out data set. This concept is also used to evaluate the quality of a clustering, where it is referred to as cluster validation (Ullmann et al. 2022), and may add to the diagnostic capabilities of FACT methods.

Contributions Towards Research Objectives. Let me now briefly summarize how Chapters 5 to 10 contribute towards this dissertation's research objectives.

Sensitivity Analysis. Working towards a change of direction in the research community to synthesize ML and SA, better reference related work, and exploit research gaps.

Contribution: Chapters 5, 6, and 7 call attention to how pervasive the principle of computing sensitivities is in ML, what pitfalls may arise due to this methodology when interpreting SL models, and how IML can be connected with related work in SA.

Marginal Effects. Adapting MEs for the application to non-linear models, thereby establishing them as a valuable model-agnostic interpretation method.

Contribution: Chapters 8 and 9 introduce a unified definition of FMEs for non-linear models, including enhancements via the NLM and the cAME, that is made accessible and extensible through an open source software implementation.

Cluster Explanations. Adapting interpretation approaches from SL to the unsupervised clustering setting.

Contribution: Chapter 10 presents a general guideline on the steps involved in algorithm-agnostic cluster explanations by the means of SA termed FACT and the two methods SMART and IDEA.

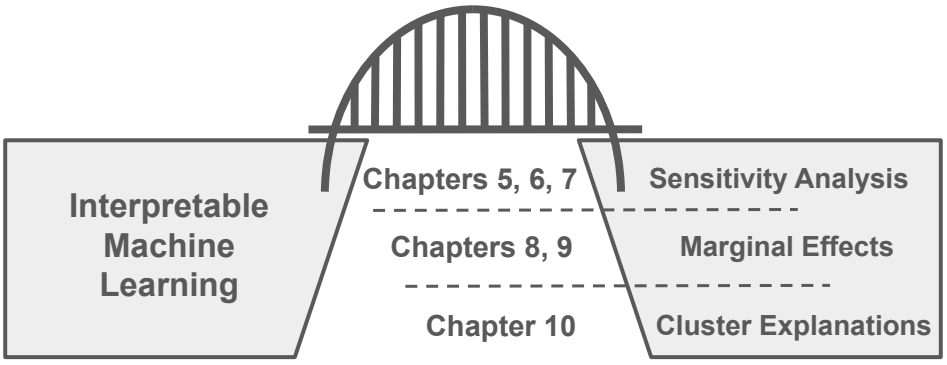


Figure 3.1.: The contributing papers in this dissertation bridge the gap between the status quo in IML and SA (Chapters 5, 6, 7), MEs (Chapters 8 and 9), and cluster explanations (Chapter 10).

4 Concluding Remarks

I would like to conclude the first part of my dissertation by discussing the broader impact of my work and future developments in IML.

According to Molnar et al. (2020), IML as an independent field of research has reached a first level of maturity, which is attributable to survey papers, (philosophical) work on the definition of interpretability, the evaluation of methods and their weaknesses, as well as software implementations; this is accompanied by a widespread adoption in the private sector, reinforced by an increasing regulatory framework that demands explainability of ML such as the General Data Protection Regulation in the European Union (Wachter et al. 2018). This raises the question how the field will (or ought to) proceed from here.

I speculate that IML will attract even more attention in the near future due to multiple reasons: ML—often referred to as AI—has a growing impact on society and will likely permeate our entire lives in the near future; furthermore, the ability to explain various aspects of ML is possibly underappreciated in many practical settings where ML is already applied; and the possibilities for explanations in ML are much more diverse than mere model interpretations, which is the most visible area of IML at the moment (Molnar 2022).

Regarding the latter, the diverse topics of this dissertation, discussed under the umbrella of interpretability, motivate us to broaden the perspective on IML. Besides HPO and cluster explanations, there are various other aspects of ML that can profit from (better) explanations. For instance, research is already carried out in interpretable reinforcement learning (Milani et al. 2023; Glanois et al. 2022) or interpretable dimensionality reduction (Björklund et al. 2023). With this dissertation, I hope to contribute towards a a more comprehensive perspective on IML.

Furthermore, I argue that the development of mathematical and possibly axiomatic frameworks for interpretations in ML is worth pursuing further; not only from a theoretical viewpoint but also a practical one, as it will contribute towards a more goal-oriented and justifiable application of techniques in real-world scenarios.

So far, I only discussed the positive impact of ML on society, but I must not fail to mention the conspicuous threats of ML to humanity as well. The rapid progress of ML with machines solving certain tasks even better than humans evokes connotations of a future where humans will be surpassed by machines in all aspects of our lives. For instance, humans have already

been surpassed in playing Go (Lee et al. 2016), and ChatGPT is now able to generate essays whose quality is rated higher than that of human-written essays (Herbold et al. 2023). In a large recent survey, 2778 top researchers in ML were asked to conjecture about the pace of progress of the field; the aggregate forecast puts the chance of machines outperforming humans in every possible task by 2047 at 50% (Grace et al. 2024). For one, this necessitates the development of ethical guidelines regulating ML, many of which have already been published (Hagendorff 2020). Beyond that, the ability to understand how the machines we train operate will be instrumental in avoiding adverse outcomes for humanity. It creates the means for humans to control the deployment of ML in real-world scenarios. If we understand how the machine is trained and how it predicts, we can make (ethically) justifiable decisions when to intervene or even decide against using ML. Trustworthy and responsible AI (Barredo Arrieta et al. 2020; Liu et al. 2022; Díaz-Rodríguez et al. 2023) are newly-emerging terms pertaining to a diverse set of desiderata of ML systems, where explainability is just one aspect among others, including fairness, privacy, or robustness. I predict that such multi-faceted umbrella terms will become increasingly relevant when discussing IML.

I therefore confidently assume that the impact of IML on society is—and will stay—an overall positive one. Through my dissertation, I hope to have had a small positive impact as well. With this closing statement, I invite the reader to continue with the contributing papers in Part II.

Bibliography

Adadi, A. and Berrada, M. (2018). "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)". In: *IEEE Access* 6, pp. 52138–52160. DOI: 10.1109/ACCESS.2018.2870052.

Angelino, E., Larus-Stone, N., Alabi, D., Seltzer, M., and Rudin, C. (2018). "Learning Certifiably Optimal Rule Lists for Categorical Data". In: *Journal of Machine Learning Research* 18.234, pp. 1–78.

Apley, D. W. and Zhu, J. (2020). "Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 82.4, pp. 1059–1086. DOI: 10.1111/rssb.12377.

Arel-Bundock, V. (2023). *marginaleffects: Predictions, Comparisons, Slopes, Marginal Means, and Hypothesis Tests*. R package version 0.17.0.9002. URL: https://marginaleffects.com.

Athey, S. and Imbens, G. W. (2019). "Machine Learning Methods That Economists Should Know About". In: *Annual Review of Economics* 11.1, pp. 685–725. DOI: 10.1146/annurev-economics-080217-053433.

Au, Q., Herbinger, J., Stachl, C., Bischl, B., and Casalicchio, G. (2022). "Grouped Feature Importance and Combined Features Effect Plot". In: *Data Mining and Knowledge Discovery* 36.4, pp. 1401–1450. DOI: 10.1007/s10618-022-00840-5.

Ballester-Ripoll, R., Paredes, E. G., and Pajarola, R. (2019). "Sobol Tensor Trains for Global Sensitivity Analysis". In: *Reliability Engineering & System Safety* 183, pp. 311–322. DOI: 10.1016/j.ress.2018.11.007.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2020). "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges Toward Responsible AI". In: *Information Fusion* 58, pp. 82–115. DOI: 10.1016/j.inffus.2019.12.012.

Baur, D., Cariboni, J., and Campolongo, F. (2004). "Global Sensitivity Analysis for Latent Factor Credit Risk Models". In: *International Journal of Risk Assessment and Management* 11. DOI: 10.2139/ssrn.638563.

Becker, W., Rowson, J., Oakley, J., Yoxall, A., Manson, G., and Worden, K. (2011). "Bayesian Sensitivity Analysis of a Model of the Aortic Valve". In: *Journal of Biomechanics* 44.8, pp. 1499–1506. DOI: 10.1016/j.jbiomech.2011.03.008.

Bertsimas, D., Orfanoudaki, A., and Wiberg, H. (2021). "Interpretable Clustering: An Optimization Approach". In: *Machine Learning* 110.1, pp. 89–138. DOI: 10.1007/s10994-020-05896-2.

Beven, K. (2012). "Predicting Hydrographs Using Models Based on Data". In: *Rainfall-Runoff Modelling*. John Wiley & Sons, Ltd. Chap. 4, pp. 83–117. DOI: 10.1002/9781119951001.ch4.

Biedenkapp, A., Lindauer, M., Eggensperger, K., Hutter, F., Fawcett, C., and Hoos, H. (2017). "Efficient Parameter Importance Analysis via Ablation with Surrogates". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 31.1. DOI: 10.1609/aaai.v31i1.10657.

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A.-L., Deng, D., and Lindauer, M. (2023). "Hyperparameter Optimization: Foundations, Algorithms, Best Practices, and Open Challenges". In: *WIREs Data Mining and Knowledge Discovery* 13.2, e1484. DOI: 10.1002/widm.1484.

Björklund, A., Mäkelä, J., and Puolamäki, K. (2023). "SLISEMAP: Supervised Dimensionality Reduction Through Local Explanations". In: *Machine Learning* 112.1, pp. 1–43. DOI: 10.1007/s10994-022-06261-1.

Borgonovo, E., Castaings, W., and Tarantola, S. (2012). "Model Emulation and Moment-Independent Sensitivity Analysis: An Application to Environmental Modelling". In: *Environmental Modelling & Software* 34, pp. 105–115. DOI: 10.1016/j.envsoft.2011.06.006.

Borgonovo, E. and Plischke, E. (2016). "Sensitivity Analysis: A Review of Recent Advances". In: *European Journal of Operational Research* 248.3, pp. 869–887. DOI: 10.1016/j.ejor.2015.06.032.

Boulesteix, A.-L., Wright, M. N., Hoffmann, S., and König, I. R. (2020). "Statistical Learning Approaches in the Genetic Epidemiology of Complex Diseases". In: *Human Genetics* 139.1, pp. 73–84. DOI: 10.1007/s00439-019-01996-9.

Breiman, L. (2001a). "Random Forests". In: *Machine Learning* 45.1, pp. 5–32. DOI: 10.1023/A:1010933404324.

– (2001b). "Statistical Modeling: The Two Cultures (with comments and a rejoinder by the author)". In: *Statistical Science* 16.3, pp. 199–231. DOI: 10.1214/ss/1009213726.

Campolongo, F., Cariboni, J., and Saltelli, A. (2007). "An Effective Screening Design for Sensitivity Analysis of Large Models". In: *Environmental Modelling and Software* 22, pp. 1509–1518.

Carrizosa, E., Kurishchenko, K., Marín, A., and Romero Morales, D. (2022). "Interpreting Clusters via Prototype Optimization". In: *Omega* 107, p. 102543. DOI: 10.1016/j.omega.2021.102543.

Casalicchio, G., Molnar, C., and Bischl, B. (2019). "Visualizing the Feature Importance for Black Box Models". In: *Machine Learning and Knowledge Discovery in Databases*. Ed. by Berlingerio, M., Bonchi, F., Gärtner, T., Hurley, N., and Ifrim, G. Cham: Springer International Publishing, pp. 655–670.

Chen, V., Li, J., Kim, J. S., Plumb, G., and Talwalkar, A. (2022). "Interpretable Machine Learning: Moving from Mythos to Diagnostics". In: *Queue* 19.6, pp. 28–56. DOI: 10.1145/3511299.

Collins, N., Theurich, G., DeLuca, C., Suarez, M., Trayanov, A., Balaji, V., Li, P., Yang, W., Hill, C., and Silva, A. da (2005). "Design and Implementation of Components in the Earth System Modeling Framework". In: *The International Journal of High Performance Computing Applications* 19.3, pp. 341–350. DOI: 10.1177/1094342005056120.

Covert, I. C., Lundberg, S., and Lee, S.-I. (2020). "Understanding Global Feature Contributions with Additive Importance Measures". In: *Proceedings of the 34th International Conference on Neural Information Processing Systems*. NIPS'20. Vancouver, BC, Canada: Curran Associates Inc.

Danilevsky, M., Qian, K., Aharonov, R., Katsis, Y., Kawas, B., and Sen, P. (2020). "A Survey of the State of Explainable AI for Natural Language Processing". In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. Ed. by Wong, K.-F., Knight, K., and Wu, H. Suzhou, China: Association for Computational Linguistics, pp. 447–459.

Das, D., Banerjee, S., and Chernova, S. (2021). "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*. HRI '21. Boulder, CO, USA: Association for Computing Machinery, pp. 351–360. DOI: 10.1145/3434073.3444657.

De Koninck, P., De Weerdt, J., and Broucke, S. K. L. M. vanden (2017). "Explaining Clusterings of Process Instances". In: *Data Mining and Knowledge Discovery* 31.3, pp. 774–808. DOI: 10.1007/s10618-016-0488-4.

Deeks, A. (2019). "The Judicial Demand For Explainable Artificial Intelligence". In: *Columbia Law Review* 119.7, pp. 1829–1850.

Dell'Oca, A., Riva, M., and Guadagnini, A. (2017). "Moment-Based Metrics for Global Sensitivity Analysis of Hydrological Systems". In: *Hydrology and Earth System Sciences* 21.12, pp. 6219–6234. DOI: 10.5194/hess-21-6219-2017.

Deng, L. and Li, X. (2013). "Machine Learning Paradigms for Speech Recognition: An Overview". In: *IEEE Transactions on Audio, Speech, and Language Processing* 21.5, pp. 1060–1089. DOI: 10.1109/TASL.2013.2244083.

Díaz-Rodríguez, N., Del Ser, J., Coeckelbergh, M., López de Prado, M., Herrera-Viedma, E., and Herrera, F. (2023). "Connecting the Dots in Trustworthy Artificial Intelligence: From AI Principles, Ethics, and Key Requirements to Responsible AI Systems and Regulation". In: *Information Fusion* 99, p. 101896. DOI: 10.1016/j.inffus.2023.101896.

Dueben, P. D. and Bauer, P. (2018). "Challenges and Design Choices for Global Weather and Climate Models Based on Machine Learning". In: *Geoscientific Model Development* 11.10, pp. 3999–4009. DOI: 10.5194/gmd-11-3999-2018.

Dwyer, D. B., Falkai, P., and Koutsouleris, N. (2018). "Machine Learning Approaches for Clinical Psychology and Psychiatry". In: *Annual Review of Clinical Psychology* 14.1, pp. 91–118. DOI: 10.1146/annurev-clinpsy-032816-045037.

Ellis, C. A., Sendi, M. S. E., Geenjaar, E. P. T., Plis, S. M., Miller, R. L., and Calhoun, V. D. (2021). "Algorithm-Agnostic Explainability for Unsupervised Clustering". In: arXiv: 2105.08053 [cs.LG].

Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD'96. Portland, Oregon: AAAI Press, pp. 226–231.

Evans, T., Retzlaff, C. O., Geißler, C., Kargl, M., Plass, M., Müller, H., Kiehl, T.-R., Zerbe, N., and Holzinger, A. (2022). "The Explainability Paradox: Challenges for xAI in Digital Pathology". In: *Future Generation Computer Systems* 133, pp. 281–296. DOI: 10.1016/j.future.2022.03.009.

Ezugwu, A. E., Ikotun, A. M., Oyelade, O. O., Abualigah, L., Agushaka, J. O., Eke, C. I., and Akinyelu, A. A. (2022). "A Comprehensive Survey of Clustering Algorithms: State-of-the-Art Machine Learning Applications, Taxonomy, Challenges, and Future Research Prospects". In: *Engineering Applications of Artificial Intelligence* 110, p. 104743. DOI: 10.1016/j.engappai.2022.104743.

Fawcett, C. and Hoos, H. H. (2016). "Analysing Differences Between Algorithm Configurations Through Ablation". In: *Journal of Heuristics* 22.4, pp. 431–458. DOI: 10.1007/s10732-014-9275-9.

Fellous, J.-M., Sapiro, G., Rossi, A., Mayberg, H., and Ferrante, M. (2019). "Explainable Artificial Intelligence for Neuroscience: Behavioral Neurostimulation". In: *Frontiers in Neuroscience* 13. DOI: 10.3389/fnins.2019.01346.

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". In: *Journal of Machine Learning Research* 20.177, pp. 1–81.

Fraiman, R., Ghattas, B., and Svarc, M. (2013). "Interpretable Clustering Using Unsupervised Binary Trees". In: *Adv. Data Anal. Classif.* 7.2, pp. 125–145. DOI: 10.1007/s11634-013-0129-3.

Freiesleben, T. and König, G. (2023). "Dear XAI Community, We Need to Talk!" In: *Explainable Artificial Intelligence: First World Conference, xAI 2023, Lisbon, Portugal, July 26–28, 2023, Proceedings, Part I*. Ed. by Longo, L. Communications in Computer and Information Science, vol 1901. Cham: Springer Nature Switzerland, pp. 48–65.

Freiesleben, T., König, G., Molnar, C., and Tejero-Cantero, Á. (2024). "Scientific Inference with Interpretable Machine Learning: Analyzing Models to Learn About Real-World Phenomena". In: *Minds and Machines* 34.3, p. 32. DOI: 10.1007/s11023-024-09691-z.

Friedman, J. H. (1991). "Multivariate Adaptive Regression Splines". In: *The Annals of Statistics* 19.1, pp. 1–67. DOI: 10.1214/aos/1176347963.

– (2001). "Greedy Function Approximation: A Gradient Boosting Machine." In: *Ann. Statist.* 29.5, pp. 1189–1232. DOI: 10.1214/aos/1013203451.

Gao, Y., Sahin, A., and Vrugt, J. A. (2023). "Probabilistic Sensitivity Analysis With Dependent Variables: Covariance-Based Decomposition of Hydrologic Models". In: *Water Resources Research* 59.4, e2022WR032834. DOI: 10.1029/2022WR032834.

Gevaert, C. M. (2022). "Explainable AI for Earth Observation: A Review Including Societal and Regulatory Perspectives". In: *International Journal of Applied Earth Observation and Geoinformation* 112, p. 102869. DOI: 10.1016/j.jag.2022.102869.

Glanois, C., Weng, P., Zimmer, M., Li, D., Yang, T., Hao, J., and Liu, W. (2022). "A Survey on Interpretable Reinforcement Learning". In: arXiv: 2112.13112 [cs.LG].

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2015). "Peeking Inside the Black Box: Visualizing Statistical Learning With Plots of Individual Conditional Expectation". In: *Journal of Computational and Graphical Statistics* 24.1, pp. 44–65. DOI: 10.1080/10618600.2014.907095.

Goodell, J. W., Kumar, S., Lim, W. M., and Pattnaik, D. (2021). "Artificial Intelligence and Machine Learning in Finance: Identifying Foundations, Themes, and Research Clusters from Bibliometric Analysis". In: *Journal of Behavioral and Experimental Finance* 32, p. 100577. DOI: 10.1016/j.jbef.2021.100577.

Goodman, B. and Flaxman, S. (2017). "European Union Regulations on Algorithmic Decision-Making and a 'Right to Explanation'". In: *AI Magazine* 38.3, pp. 50–57. DOI: 10.1609/aimag.v38i3.2741.

Gordon, L., Grantcharov, T., and Rudzicz, F. (2019). "Explainable Artificial Intelligence for Safe Intraoperative Decision Support". In: *JAMA Surgery* 154.11, pp. 1064–1065. DOI: 10.1001/jamasurg.2019.2821.

Grace, K., Stewart, H., Sandkühler, J. F., Thomas, S., Weinstein-Raun, B., and Brauner, J. (2024). "Thousands of AI Authors on the Future of AI". In: arXiv: 2401.02843 [cs.CY].

Greene, W. (2019). *Econometric Analysis*. 8th ed. Pearson International.

Guo, L., Meng, Z., Sun, Y., and Wang, L. (2016). "Parameter Identification and Sensitivity Analysis of Solar Cell Models with Cat Swarm Optimization Algorithm". In: *Energy Conversion and Management* 108, pp. 520–528. DOI: 10.1016/j.enconman.2015.11.041.

Gupta, H. V. and Razavi, S. (2018). "Revisiting the Basis of Sensitivity Analysis for Dynamical Earth System Models". In: *Water Resources Research* 54.11, pp. 8692–8717. DOI: 10.1029/2018WR022668.

Hagendorff, T. (2020). "The Ethics of AI Ethics: An Evaluation of Guidelines". In: *Minds and Machines* 30.1, pp. 99–120. DOI: 10.1007/s11023-020-09517-8.

Haghnegahdar, A. and Razavi, S. (2017). "Insights Into Sensitivity Analysis of Earth and Environmental Systems Models: On the Impact of Parameter Perturbation Scale". In: *Environmental Modelling & Software* 95, pp. 115–131. DOI: 10.1016/j.envsoft.2017.03.031.

Han, J., Kamber, M., and Pei, J. (2012). "10 - Cluster Analysis: Basic Concepts and Methods". In: *Data Mining*. Ed. by Han, J., Kamber, M., and Pei, J. 3rd ed. The Morgan Kaufmann

Series in Data Management Systems. Boston: Morgan Kaufmann, pp. 443–495. DOI: 10.1016/B978-0-12-381479-1.00010-1.

Harenberg, D., Marelli, S., Sudret, B., and Winschel, V. (2019). "Uncertainty Quantification and Global Sensitivity Analysis for Economic Models". In: *Quantitative Economics* 10.1, pp. 1–41. DOI: 10.3982/QE866.

Hempel, C. G. and Oppenheim, P. (1948). "Studies in the Logic of Explanation". In: *Philosophy of Science* 15.2, pp. 135–175. DOI: 10.1086/286983.

Herbold, S., Hautli-Janisz, A., Heuer, U., Kikteva, Z., and Trautsch, A. (2023). "A Large-Scale Comparison of Human-Written Versus ChatGPT-Generated Essays". In: *Scientific Reports* 13.1, p. 18617. DOI: 10.1038/s41598-023-45644-9.

Hill, C., DeLuca, C., Balaji, V., Suarez, M., and Silva, A. d. (2004). "The Architecture of the Earth System Modeling Framework". In: *Computing in Science and Engineering* 6.1, pp. 18–28. DOI: 10.1109/MCISE.2004.1255817.

Hutter, F., Hoos, H., and Leyton-Brown, K. (2014). "An Efficient Approach for Assessing Hyperparameter Importance". In: *Proceedings of the 31st International Conference on Machine Learning*. Ed. by Xing, E. P. and Jebara, T. Vol. 32. Proceedings of Machine Learning Research 1. Bejing, China: PMLR, pp. 754–762.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2011). "Sequential Model-Based Optimization for General Algorithm Configuration". In: *Learning and Intelligent Optimization - 5th International Conference, LION 5, Rome, Italy, January 17-21, 2011. Selected Papers*. Ed. by Coello, C. A. C. Vol. 6683. Lecture Notes in Computer Science. Springer, pp. 507–523. DOI: 10.1007/978-3-642-25566-3_40.

Iooss, B. and Lemaître, P. (2015). "A Review on Global Sensitivity Analysis Methods". In: *Uncertainty Management in Simulation-Optimization of Complex Systems: Algorithms and Applications*. Ed. by Dellino, G. and Meloni, C. Boston, MA: Springer US, pp. 101–122. DOI: 10.1007/978-1-4899-7547-8_5.

Jain, A. K. (2010). "Data Clustering: 50 Years Beyond K-Means". In: *Pattern Recognition Letters* 31.8, pp. 651–666. DOI: 10.1016/j.patrec.2009.09.011.

Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Data*. USA: Prentice-Hall, Inc.

Janzing, D., Minorics, L., and Bloebaum, P. (2020). "Feature Relevance Quantification in Explainable AI: A Causal Problem". In: *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*. Ed. by Chiappa, S. and Calandra, R. Vol. 108. Proceedings of Machine Learning Research. PMLR, pp. 2907–2916.

Jiménez-Luna, J., Grisoni, F., and Schneider, G. (2020). "Drug Discovery with Explainable Artificial Intelligence". In: *Nature Machine Intelligence* 2.10, pp. 573–584. DOI: 10.1038/s42256-020-00236-4.

Jones, D. R., Schonlau, M., and Welch, W. J. (1998). "Efficient Global Optimization of Expensive Black-Box Functions". In: *J. Glob. Optim.* 13.4, pp. 455–492. DOI: 10.1023/A:1008306431147.

Kamath, U. and Liu, J. (2021). "Introduction to Interpretability and Explainability". In: *Explainable Artificial Intelligence: An Introduction to Interpretable Machine Learning*. Cham: Springer International Publishing, pp. 1–26. DOI: 10.1007/978-3-030-83356-5_1.

Khosravi, H., Shum, S. B., Chen, G., Conati, C., Tsai, Y.-S., Kay, J., Knight, S., Martinez-Maldonado, R., Sadiq, S., and Gašević, D. (2022). "Explainable Artificial Intelligence in Education". In: *Computers and Education: Artificial Intelligence* 3, p. 100074. DOI: 10.1016/j.caeai.2022.100074.

Kim, B., Khanna, R., and Koyejo, O. O. (2016). "Examples Are Not Enough, Learn to Criticize! Criticism for Interpretability". In: *Advances in Neural Information Processing Systems*. Ed. by Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. Vol. 29. Curran Associates, Inc.

Krishnan, M. (2020). "Against Interpretability: A Critical Examination of the Interpretability Problem in Machine Learning". In: *Philosophy & Technology* 33.3, pp. 487–502. DOI: 10.1007/s13347-019-00372-9.

Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., Au, Q., Casalicchio, G., Kotthoff, L., and Bischl, B. (2019). "mlr3: A Modern Object-Oriented Machine Learning Framework in R". In: *Journal of Open Source Software*. DOI: 10.21105/joss.01903.

Lawless, C. and Gunluk, O. (2023). "Cluster Explanation via Polyhedral Descriptions". In: *Proceedings of the 40th International Conference on Machine Learning*. Ed. by Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. Vol. 202. Proceedings of Machine Learning Research. PMLR, pp. 18652–18666.

Lawless, C., Kalagnanam, J., Nguyen, L. M., Phan, D., and Reddy, C. (2022). "Interpretable Clustering via Multi-Polytope Machines". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 36.7, pp. 7309–7316. DOI: 10.1609/aaai.v36i7.20693.

Le Gratiet, L., Marelli, S., and Sudret, B. (2017). "Metamodel-Based Sensitivity Analysis: Polynomial Chaos Expansions and Gaussian Processes". In: *Handbook of Uncertainty Quantification*. Cham: Springer International Publishing, pp. 1289–1325. DOI: 10.1007/978-3-319-12385-1_38.

Lee, C.-S., Wang, M.-H., Yen, S.-J., Wei, T.-H., Wu, I.-C., Chou, P.-C., Chou, C.-H., Wang, M.-W., and Yan, T.-H. (2016). "Human vs. Computer Go: Review and Prospect [Discussion Forum]". In: *IEEE Computational Intelligence Magazine* 11.3, pp. 67–72. DOI: 10.1109/MCI.2016.2572559.

Lipton, Z. C. (2018). "The Mythos of Model Interpretability: In Machine Learning, the Concept of Interpretability Is Both Important and Slippery." In: *Queue* 16.3, pp. 31–57. DOI: 10.1145/3236386.3241340.

Liu, H., Wang, Y., Fan, W., Liu, X., Li, Y., Jain, S., Liu, Y., Jain, A., and Tang, J. (2022). "Trustworthy AI: A Computational Perspective". In: *ACM Trans. Intell. Syst. Technol.* 14.1. DOI: 10.1145/3546872.

Löwe, H., Scholbeck, C. A., Heumann, C., Bischl, B., and Casalicchio, G. (2023). "fmeffects: An R Package for Forward Marginal Effects". In: arXiv: 2310.02008 [cs.LG].

Lundberg, S. M. and Lee, S.-I. (2017). "A Unified Approach to Interpreting Model Predictions". In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS'17. Long Beach, California, USA: Curran Associates Inc., pp. 4768–4777.

Machlev, R., Heistrene, L., Perl, M., Levy, K., Belikov, J., Mannor, S., and Levron, Y. (2022). "Explainable Artificial Intelligence (XAI) Techniques for Energy and Power Systems: Review, Challenges and Opportunities". In: *Energy and AI* 9, p. 100169. DOI: 10.1016/j.egyai.2022.100169.

MacQueen, J. (1967). "Some Methods for Classification and Analysis of Multivariate Observations". In: *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability - Vol. 1*. Ed. by Le Cam, L. M. and Neyman, J. University of California Press, Berkeley, CA, USA, pp. 281–297.

Mai, J., Craig, J. R., Tolson, B. A., and Arsenault, R. (2022). "The Sensitivity of Simulated Streamflow to Individual Hydrologic Processes Across North America". In: *Nature Communications* 13.1, p. 455. DOI: 10.1038/s41467-022-28010-7.

Marcinkevičs, R. and Vogt, J. E. (2023). "Interpretability and Explainability: A Machine Learning Zoo Mini-Tour". In: arXiv: 2012.01805 [cs.LG].

Marrel, A., Iooss, B., Laurent, B., and Roustant, O. (2009). "Calculations of Sobol Indices for the Gaussian Process Metamodel". In: *Reliability Engineering & System Safety* 94, pp. 742–751. DOI: 10.1016/j.ress.2008.07.008.

Marrel, A., Iooss, B., Van Dorpe, F., and Volkova, E. (2008). "An Efficient Methodology for Modeling Complex Computer Codes with Gaussian Processes". In: *Computational Statistics & Data Analysis* 52, pp. 4731–4744. DOI: 10.1016/j.csda.2008.03.026.

McCabe, C. J., Halvorson, M. A., King, K. M., Cao, X., and Kim, D. S. (2022). "Interpreting Interaction Effects in Generalized Linear Models of Nonlinear Probabilities and Counts". In: *Multivariate Behavioral Research* 57.2-3, pp. 243–263. DOI: 10.1080/00273171.2020.1868966.

Milani, S., Topin, N., Veloso, M., and Fang, F. (2023). "Explainable Reinforcement Learning: A Survey and Comparative Review". In: *ACM Comput. Surv.* DOI: 10.1145/3616864.

Miller, T. (2019). "Explanation in Artificial Intelligence: Insights From the Social Sciences". In: *Artificial Intelligence* 267, pp. 1–38. DOI: 10.1016/j.artint.2018.07.007.

Molnar, C. (2022). *Interpretable Machine Learning: A Guide for Making Black Box Models Explainable*. 2nd ed. URL: https://christophm.github.io/interpretable-ml-book.

Molnar, C., Casalicchio, G., and Bischl, B. (2020). "Interpretable Machine Learning – A Brief History, State-of-the-Art and Challenges". In: *ECML PKDD 2020 Workshops*. Ed. by Koprinska, I., Kamp, M., Appice, A., Loglisci, C., Antonie, L., Zimmermann, A., Guidotti, R., Özgöbek, Ö., Ribeiro, R. P., Gavaldà, R., Gama, J., Adilova, L., Krishnamurthy, Y., Ferreira, P. M., Malerba, D., Medeiros, I., Ceci, M., Manco, G., Masciari, E., Ras, Z. W., Christen, P., Ntoutsi, E., Schubert, E., Zimek, A., Monreale, A., Biecek, P., Rinzivillo, S., Kille, B., Lommatzsch, A., and Gulla, J. A. Cham: Springer International Publishing, pp. 417–431.

Molnar, C. and Freiesleben, T. (2024a). *Supervised Machine Learning For Science: How to Stop Worrying and Love Your Black Box*. URL: https://ml-science-book.com/.

Molnar, C., König, G., Bischl, B., and Casalicchio, G. (2024b). "Model-Agnostic Feature Importance and Effects with Dependent Features: A Conditional Subgroup Approach". In: *Data Mining and Knowledge Discovery* 38.5, pp. 2903–2941. DOI: 10.1007/s10618-022-00901-9.

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: *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*. Ed. by Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.-R., and Samek, W. Vol. 13200. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 39–68. DOI: 10.1007/978-3-031-04083-2_4.

Moosbauer, J., Herbinger, J., Casalicchio, G., Lindauer, M., and Bischl, B. (2021). "Explaining Hyperparameter Optimization via Partial Dependence Plots". In: *Advances in Neural Information Processing Systems*. Ed. by Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. URL: https://openreview.net/forum?id=k8KDqVbIS2l.

Morris, M. D. (1991). "Factorial Sampling Plans for Preliminary Computational Experiments". In: *Technometrics* 33.2, pp. 161–174.

Mullainathan, S. and Spiess, J. (2017). "Machine Learning: An Applied Econometric Approach". In: *Journal of Economic Perspectives* 31.2, pp. 87–106. DOI: 10.1257/jep.31.2.87.

Naz, B. S., Sharples, W., Ma, Y., Goergen, K., and Kollet, S. (2023). "Continental-Scale Evaluation of a Fully Distributed Coupled Land Surface and Groundwater Model, ParFlow-CLM (v3.6.0), Over Europe". In: *Geoscientific Model Development* 16.6, pp. 1617–1639. DOI: 10.5194/gmd-16-1617-2023.

Nossent, J., Elsen, P., and Bauwens, W. (2011). "Sobol' Sensitivity Analysis of a Complex Environmental Model". In: *Environmental Modelling & Software* 26.12, pp. 1515–1525. DOI: 10.1016/j.envsoft.2011.08.010.

Onukwugha, E., Bergtold, J., and Jain, R. (2015). "A Primer on Marginal Effects—Part I: Theory and Formulae". In: *PharmacoEconomics* 33.1, pp. 25–30. DOI: 10.1007/s40273-014-0210-6.

Pierson, H. A. and Gashler, M. S. (2017). "Deep Learning in Robotics: A Review of Recent Research". In: *Advanced Robotics* 31.16, pp. 821–835. DOI: 10.1080/01691864.2017.1365009.

Plant, C. and Böhm, C. (2011). "INCONCO: Interpretable Clustering of Numerical and Categorical Objects". In: *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '11. San Diego, California, USA: Association for Computing Machinery, pp. 1127–1135. DOI: 10.1145/2020408.2020584.

Rabitz, H. and Aliş, Ö. F. (1999). "General Foundations of High-Dimensional Model Representations". In: *Journal of Mathematical Chemistry* 25.2, pp. 197–233. DOI: 10.1023/A:1019188517934.

Rajkomar, A., Dean, J., and Kohane, I. (2019). "Machine Learning in Medicine". In: *New England Journal of Medicine* 380.14, pp. 1347–1358. DOI: 10.1056/NEJMra1814259.

Ratto, M. (2008). "Analysing DSGE Models with Global Sensitivity Analysis". In: *Computational Economics* 31.2, pp. 115–139. DOI: 10.1007/s10614-007-9110-6.

Razavi, S. (2021). "Deep Learning, Explained: Fundamentals, Explainability, and Bridgeability to Process-Based Modelling". In: *Environmental Modelling & Software* 144, p. 105159. DOI: 10.1016/j.envsoft.2021.105159.

Razavi, S. and Gupta, H. V. (2015). "What Do We Mean by Sensitivity Analysis? The Need for Comprehensive Characterization of "Global" Sensitivity in Earth and Environmental Systems Models". In: *Water Resources Research* 51.5, pp. 3070–3092. DOI: 10.1002/2014WR016527.

– (2016). "A New Framework for Comprehensive, Robust, and Efficient Global Sensitivity Analysis: 1. Theory". In: *Water Resources Research* 52.1, pp. 423–439. DOI: 10.1002/2015WR017558.

Razavi, S., Jakeman, A., Saltelli, A., Prieur, C., Iooss, B., Borgonovo, E., Plischke, E., Lo Piano, S., Iwanaga, T., Becker, W., Tarantola, S., Guillaume, J. H., Jakeman, J., Gupta, H., Melillo, N., Rabitti, G., Chabridon, V., Duan, Q., Sun, X., Smith, S., Sheikholeslami, R., Hosseini, N., Asadzadeh, M., Puy, A., Kucherenko, S., and Maier, H. R. (2021). "The Future of Sensitivity Analysis: An Essential Discipline for Systems Modeling and Policy Support". In: *Environmental Modelling & Software* 137, p. 104954. DOI: 10.1016/j.envsoft.2020.104954.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., and Prabhat (2019). "Deep Learning and Process Understanding for Data-Driven Earth System Science". In: *Nature* 566.7743, pp. 195–204. DOI: 10.1038/s41586-019-0912-1.

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*. KDD '16. San Francisco, California, USA: Association for Computing Machinery, pp. 1135–1144. DOI: 10.1145/2939672.2939778.

– (2018). "Anchors: High-Precision Model-Agnostic Explanations". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1. DOI: 10.1609/aaai.v32i1.11491.

Rousseeuw, P. J. (1987). "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis". In: *Journal of Computational and Applied Mathematics* 20, pp. 53–65. DOI: 10.1016/0377-0427(87)90125-7.

Rudin, C. (2019). "Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead". In: *Nature Machine Intelligence* 1.5, pp. 206–215. DOI: 10.1038/s42256-019-0048-x.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. (2022). "Interpretable Machine Learning: Fundamental Principles and 10 Grand Challenges". In: *Statistics Surveys* 16, pp. 1–85. DOI: 10.1214/21-SS133.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008). *Global Sensitivity Analysis: The Primer*. John Wiley & Sons Ltd.

Saltelli, A. and Annoni, P. (2010). "How to Avoid a Perfunctory Sensitivity Analysis". In: *Environmental Modelling & Software* 25.12, pp. 1508–1517. DOI: 10.1016/j.envsoft.2010.04.012.

Saltelli, A. and Tarantola, S. (2002). "On the Relative Importance of Input Factors in Mathematical Models". In: *Journal of the American Statistical Association* 97.459, pp. 702–709. DOI: 10.1198/016214502388618447.

Samuel, A. L. (1959). "Some Studies in Machine Learning Using the Game of Checkers". In: *IBM J. Res. Dev.* 3.3, pp. 210–229. DOI: 10.1147/rd.33.0210.

Santner, T. J., Williams, B. J., and Notz, W. I. (2018). "Sensitivity Analysis and Variable Screening". In: *The Design and Analysis of Computer Experiments*. New York, NY: Springer New York, pp. 247–297. DOI: 10.1007/978-1-4939-8847-1_7.

Sauer, T. (2011). *Numerical Analysis*. 2nd ed. USA: Addison-Wesley Publishing Company.

Schmidhuber, J. (2015). "Deep Learning in Neural Networks: An Overview". In: *Neural Networks* 61, pp. 85–117. DOI: 10.1016/j.neunet.2014.09.003.

Scholbeck, C. A., Casalicchio, G., Molnar, C., Bischl, B., and Heumann, C. (2024). "Marginal Effects for Non-Linear Prediction Functions". In: *Data Mining and Knowledge Discovery* 38.5, pp. 2997–3042. DOI: 10.1007/s10618-023-00993-x.

Scholbeck, C. A., Funk, H., and Casalicchio, G. (2023a). "Algorithm-Agnostic Feature Attributions for Clustering". In: *Explainable Artificial Intelligence: First World Conference, xAI 2023, Lisbon, Portugal, July 26–28, 2023, Proceedings, Part I*. Ed. by Longo, L. Vol. 1901. Communications in Computer and Information Science. Cham: Springer Nature Switzerland, pp. 217–240. DOI: 10.1007/978-3-031-44064-9_13.

Scholbeck, C. A., Molnar, C., Heumann, C., Bischl, B., and Casalicchio, G. (2020). "Sampling, Intervention, Prediction, Aggregation: A Generalized Framework for Model-Agnostic Interpretations". In: *Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part I*. Ed. by Cellier, P. and Driessens, K. Vol. 1167. Communications in Computer and Information Science. Cham: Springer International Publishing, pp. 205–216. DOI: 10.1007/978-3-030-43823-4_18.

Scholbeck, C. A., Moosbauer, J., Casalicchio, G., Gupta, H., Bischl, B., and Heumann, C. (2023b). "Position Paper: Bridging the Gap Between Machine Learning and Sensitivity Analysis". In: arXiv: 2312.13234 [cs.LG].

Sharma, D. K., Mishra, J., Singh, A., Govil, R., Srivastava, G., and Lin, J. C.-W. (2022). "Explainable Artificial Intelligence for Cybersecurity". In: *Computers and Electrical Engineering* 103, p. 108356. DOI: 10.1016/j.compeleceng.2022.108356.

Shin, M.-J., Guillaume, J. H., Croke, B. F., and Jakeman, A. J. (2013). "Addressing Ten Questions About Conceptual Rainfall–Runoff Models with Global Sensitivity Analyses in R". In: *Journal of Hydrology* 503, pp. 135–152. DOI: 10.1016/j.jhydrol.2013.08.047.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Driessche, G. van den, Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. (2016). "Mastering the Game of Go with Deep Neural Networks and Tree Search". In: *Nature* 529.7587, pp. 484–489. DOI: 10.1038/nature16961.

Skinner, G. and Walmsley, T. (2019). "Artificial Intelligence and Deep Learning in Video Games A Brief Review". In: *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pp. 404–408. DOI: 10.1109/CCOMS.2019.8821783.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). "Practical Bayesian Optimization of Machine Learning Algorithms". In: *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'12. Lake Tahoe, Nevada: Curran Associates Inc., pp. 2951–2959.

Sobol, I. (1990). "On Sensitivity Estimation for Nonlinear Mathematical Models". In: *Matem. Mod.* 2 (1), pp. 112–118.

Sobol, I. and Kucherenko, S. (2010). "Derivative Based Global Sensitivity Measures". In: *Procedia - Social and Behavioral Sciences* 2.6. Sixth International Conference on Sensitivity Analysis of Model Output, pp. 7745–7746. DOI: 10.1016/j.sbspro.2010.05.208.

Song, X., Zhang, J., Zhan, C., Xuan, Y., Ye, M., and Xu, C. (2015). "Global Sensitivity Analysis in Hydrological Modeling: Review of Concepts, Methods, Theoretical Framework, and Applications". In: *Journal of Hydrology* 523, pp. 739–757. DOI: 10.1016/j.jhydrol.2015.02.013.

Štrumbelj, E. and Kononenko, I. (2010). "An Efficient Explanation of Individual Classifications Using Game Theory". In: *Journal of Machine Learning Research* 11.1, pp. 1–18.

Sudret, B. (2008). "Global Sensitivity Analysis Using Polynomial Chaos Expansion". In: *Reliability Engineering & System Safety* 93, pp. 964–979. DOI: 10.1016/j.ress.2007.04.002.

Sumner, T., Shephard, E., and Bogle, I. D. L. (2012). "A Methodology for Global-Sensitivity Analysis of Time-Dependent Outputs in Systems Biology Modelling". In: *J R Soc Interface* 9.74, pp. 2156–2166.

Sundararajan, M., Taly, A., and Yan, Q. (2017). "Axiomatic Attribution for Deep Networks". In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. ICML'17. Sydney, NSW, Australia: JMLR.org, pp. 3319–3328.

Tantithamthavorn, C. K. and Jiarpakdee, J. (2021). "Explainable AI for Software Engineering". In: *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pp. 1–2. DOI: 10.1109/ASE51524.2021.9678580.

Thorndike, R. L. (1953). "Who Belongs in the Family?" In: *Psychometrika* 18.4, pp. 267–276. DOI: 10.1007/BF02289263.

Tian, W. (2013). "A Review of Sensitivity Analysis Methods in Building Energy Analysis". In: *Renewable and Sustainable Energy Reviews* 20, pp. 411–419. DOI: 10.1016/j.rser.2012.12.014.

Tomašev, N. and Radovanović, M. (2016). "Clustering Evaluation in High-Dimensional Data". In: *Unsupervised Learning Algorithms*. Ed. by Celebi, M. E. and Aydin, K. Cham: Springer International Publishing, pp. 71–107. DOI: 10.1007/978-3-319-24211-8_4.

Ullmann, T., Hennig, C., and Boulesteix, A.-L. (2022). "Validation of Cluster Analysis Results on Validation Data: A Systematic Framework". In: *WIREs Data Mining and Knowledge Discovery* 12.3, e1444. DOI: 10.1002/widm.1444.

Vapnik, V. N. and Chervonenkis, A. Y. (1974). *Theory of Pattern Recognition [in Russian]*. USSR: Nauka.

Wachter, S., Mittelstadt, B., and Russell, C. (2018). "Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR". In: *Harvard Journal of Law and Technology* 31.2, pp. 841–887.

Wagener, T. and Pianosi, F. (2019). "What Has Global Sensitivity Analysis Ever Done For Us? A Systematic Review to Support Scientific Advancement and to Inform Policy-Making in Earth System Modelling". In: *Earth-Science Reviews* 194, pp. 1–18. DOI: 10.1016/j.earscirev.2019.04.006.

Wang, W., Yang, J., and Muntz, R. R. (1997). "STING: A Statistical Information Grid Approach to Spatial Data Mining". In: *Proceedings of the 23rd International Conference on Very Large Data Bases*. VLDB '97. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 186–195.

Williams, R. (2012). "Using the Margins Command to Estimate and Interpret Adjusted Predictions and Marginal Effects". In: *The Stata Journal* 12.2, pp. 308–331. DOI: 10.1177/1536867X1201200209.

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

Xu, D. and Tian, Y. (2015). "A Comprehensive Survey of Clustering Algorithms". In: *Annals of Data Science* 2.2, pp. 165–193. DOI: 10.1007/s40745-015-0040-1.

Zhang, R., Liu, Y., and Sun, H. (2020). "Physics-Informed Multi-LSTM Networks for Metamodeling of Nonlinear Structures". In: *Computer Methods in Applied Mechanics and Engineering* 369, p. 113226. DOI: 10.1016/j.cma.2020.113226.

Zhou, Y., Booth, S., Ribeiro, M. T., and Shah, J. (2022). "Do Feature Attribution Methods Correctly Attribute Features?" In: *Proceedings of the AAAI Conference on Artificial Intelligence* 36.9, pp. 9623–9633. DOI: 10.1609/aaai.v36i9.21196.

Žižka, J., Dařena, F., and Svoboda, A. (2021). *Text Mining with Machine Learning*. CRC Press. DOI: 10.1201/9780429469275.

Part II.

Contributing Papers

5 Sampling, Intervention, Prediction, Aggregation: A Generalized Framework for Model-Agnostic Interpretations

Contributing Paper

Scholbeck, C. A., Molnar, C., Heumann, C., Bischl, B., and Casalicchio, G. (2020). "Sampling, Intervention, Prediction, Aggregation: A Generalized Framework for Model-Agnostic Interpretations". In: *Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019, Würzburg, Germany, September 16–20, 2019, Proceedings, Part I*. Ed. by Cellier, P. and Driessens, K. Vol. 1167. Communications in Computer and Information Science. Cham: Springer International Publishing, pp. 205–216. DOI: 10.1007/978-3-030-43823-4_18

Copyright Notice

Declaration of Contributions

C.A. Scholbeck contributed to this paper as the first author. The paper builds upon the master thesis of C.A. Scholbeck[1], which was supervised by C. Heumann and G. Casalicchio under close academic guidance and supervision. In a chapter of his master thesis, C.A. Scholbeck demonstrated that the feature effect methods ALE, the PD, and the AME are based on the SIPA work stages. The paper extends the master thesis in multiple directions: For feature effects, it contains improved illustrations for ALE, the PD, and the AME and novel ones for LIME and Shapley values; furthermore, it contains novel illustrations for feature importance computations including the PFI, the variance of the PD, and the Shapley feature importance. C.A. Scholbeck demonstrated how each method can be deconstructed into the SIPA work

[1] Scholbeck, Christian Alexander (2018): Interpretierbares Machine-Learning. Post-hoc modellagnostische Verfahren zur Bestimmung von Prädiktoreffekten in Supervised-Learning-Modellen. Master Thesis, Ludwig-Maximilians-Universität München

stages, created all visualizations, drafted the paper, and revised it according to the feedback from his co-authors and external reviewers.

C. Molnar developed the initial idea for the SIPA framework with additional feedback from G. Casalicchio. C. Molnar, C. Heumann, B. Bischl, and G. Casalicchio assisted in revising the paper and suggested several notable modifications.

Sampling, Intervention, Prediction, Aggregation: A Generalized Framework for Model-Agnostic Interpretations

Christian A. Scholbeck(✉), Christoph Molnar, Christian Heumann, Bernd Bischl, and Giuseppe Casalicchio

Department of Statistics, Ludwig-Maximilians-University Munich, Ludwigstr. 33, 80539 Munich, Germany
christian.scholbeck@stat.uni-muenchen.de

Abstract. Model-agnostic interpretation techniques allow us to explain the behavior of any predictive model. Due to different notations and terminology, it is difficult to see how they are related. A unified view on these methods has been missing. We present the generalized SIPA (sampling, intervention, prediction, aggregation) framework of work stages for model-agnostic interpretations and demonstrate how several prominent methods for feature effects can be embedded into the proposed framework. Furthermore, we extend the framework to feature importance computations by pointing out how variance-based and performance-based importance measures are based on the same work stages. The SIPA framework reduces the diverse set of model-agnostic techniques to a single methodology and establishes a common terminology to discuss them in future work.

Keywords: Interpretable Machine Learning · Explainable AI · Feature Effect · Feature Importance · Model-Agnostic · Partial Dependence

1 Introduction and Related Work

There has been an ongoing debate about the lacking interpretability of machine learning (ML) models. As a result, researchers have put in great efforts developing techniques to create insights into the workings of predictive black box models. Interpretable machine learning [15] serves as an umbrella term for all interpretation methods in ML. We make the following distinctions:

(i) *Feature effects or feature importance:* Feature effects indicate the direction and magnitude of change in predicted outcome due to changes in feature values. Prominent methods include the individual conditional expectation (ICE) [9] and partial dependence (PD) [8], accumulated local effects (ALE) [1], Shapley values [19] and local interpretable model-agnostic explanations (LIME) [17]. The feature importance measures the importance of a feature

P. Cellier and K. Driessens (Eds.): ECML PKDD 2019 Workshops, CCIS 1167, pp. 205–216, 2020.
https://doi.org/10.1007/978-3-030-43823-4_18

to the model behavior. This includes variance-based measures like the feature importance ranking measure (FIRM) [10,20] and performance-based measures like the permutation feature importance (PFI) [7], individual conditional importance (ICI) and partial importance (PI) curves [4], as well as the Shapley feature importance (SFIMP) [4]. Input gradients were proposed by [11] as a model-agnostic tool for both effects and importance that essentially equals marginal effects (ME) [12], which have a long tradition in statistics. They also define an average input gradient which corresponds to the average marginal effect (AME).

(ii) *Intrinsic or post-hoc interpretability:* Linear models (LM), generalized linear models (GLM), classification and regression trees (CART) or rule lists [18] are examples for intrinsically interpretable models, while random forests (RF), support vector machines (SVM), neural networks (NN) or gradient boosting (GB) models can only be interpreted post-hoc. Here, the interpretation process is detached from and takes place after the model fitting process, e.g., with the ICE, PD or ALEs.

(iii) *Model-specific or model-agnostic interpretations:* Interpreting model coefficients of GLMs or deriving a decision rule from a classification tree is a model-specific interpretation. Model-agnostic methods such as the ICE, PD or ALEs can be applied to any model.

(iv) *Local or global explanations:* Local explanations like the ICE evaluate the model behavior when predicting for one specific observation. Global explanations like the PD interpret the model for the entire input space. Furthermore, it is possible to explain model predictions for a group of observations, e.g., on intervals. In a lot of cases, local and global explanations can be transformed into one another via (dis-)aggregation, e.g., the ICE and PD.

Motivation: Research in model-agnostic interpretation methods is complicated by the variety of different notations and terminology. It turns out that deconstructing model-agnostic techniques into sequential work stages reveals striking similarities. In [14] the authors propose a unified framework for model-agnostic interpretations called SHapley Additive exPlanations (SHAP). However, the SHAP framework only considers Shapley values or variations thereof (KernelSHAP and TreeSHAP). The motivation for this research paper is to provide a more extensive survey on model-agnostic interpretation methods, to reveal similarities in their computation and to establish a framework with common terminology that is applicable to all model-agnostic techniques.

Contributions: In Sect. 4 we present the generalized SIPA (sampling, intervention, prediction, aggregation) framework of work stages for model-agnostic techniques. We proceed to demonstrate how several methods to estimate feature effects (MEs, ICE and PD, ALEs, Shapley values and LIME) can be embedded into the proposed framework. Furthermore, in Sects. 5 and 6 we extend the framework to feature importance computations by pointing out how variance-based (FIRM) and performance-based (ICI and PI, PFI and SFIMP) importance measures are based on the same work stages. By using a unified notation, we also reveal how the methods are related.

2 Notation and Preliminaries

Consider a p-dimensional feature space $\mathcal{X}_P = \mathcal{X}_1 \times \cdots \times \mathcal{X}_p$ with the feature index set $P = \{1, \ldots, p\}$ and a target space $\mathcal{Y}$. We assume an unknown functional relationship f between $\mathcal{X}_P$ and $\mathcal{Y}$. A supervised learning model $\hat{f}$ attempts to learn this relationship from an i.i.d. training sample that was drawn from the unknown probability distribution $\mathcal{F}$ with the sample space $\mathcal{X}_P \times \mathcal{Y}$. The random variables generated from the feature space are denoted by $X = (X_1, \ldots, X_p)$. The random variable generated from the target space is denoted by Y. We draw an i.i.d. sample of test data $\mathcal{D}$ with n observations from $\mathcal{F}$. The vector $x^{(i)} = (x_1^{(i)}, \ldots, x_p^{(i)}) \in \mathcal{X}_P$ corresponds to the feature values of the i-th observation that are associated with the observed target value $y^{(i)} \in \mathcal{Y}$. The vector $x_j = (x_j^{(1)}, \ldots, x_j^{(n)})^\top$ represents the realizations of X_j. The generalization error $GE(\hat{f}, \mathcal{F})$ corresponds to the expectation of the loss function $\mathcal{L}$ on unseen test data from $\mathcal{F}$ and is estimated by the average loss on $\mathcal{D}$.

$$GE(\hat{f}, \mathcal{F}) = \mathbb{E}\left[\mathcal{L}(\hat{f}(X_1, \ldots, X_p), Y)\right]$$

$$\widehat{GE}(\hat{f}, \mathcal{D}) = \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\hat{f}(x_1^{(i)}, \ldots, x_p^{(i)}), y^{(i)})$$

A variety of model-agnostic techniques is used to interpret the prediction function $\hat{f}(x_1, \ldots, x_p)$ with the sample of test data $\mathcal{D}$. We estimate the effects and importance of a subset of features with index set S ($S \subseteq P$). A vector of feature values $x \in \mathcal{X}_P$ can be partitioned into two vectors x_S and $x_{\setminus S}$ so that $x = (x_S, x_{\setminus S})$. The corresponding random variables are denoted by X_S and $X_{\setminus S}$. Given a model-agnostic technique where S only contains a single element, the corresponding notations are $X_j, X_{\setminus j}$ and $x_j, x_{\setminus j}$.

The partial derivative of the trained model $\hat{f}(x_j, x_{\setminus j})$ with respect to x_j is numerically approximated with a symmetric difference quotient [12].

$$\lim_{h \to 0} \frac{\hat{f}(x_j + h, x_{\setminus j}) - \hat{f}(x_j, x_{\setminus j})}{h} \approx \frac{\hat{f}(x_j + h, x_{\setminus j}) - \hat{f}(x_j - h, x_{\setminus j})}{2h}, \quad h > 0$$

A term of the form $\hat{f}(x_j + h, x_{\setminus j}) - \hat{f}(x_j - h, x_{\setminus j})$ is called a finite difference (FD) of predictions with respect to x_j.

$$FD_{\hat{f},j}(x_j, x_{\setminus j}) = \hat{f}(x_j + h, x_{\setminus j}) - \hat{f}(x_j - h, x_{\setminus j})$$

3 Feature Effects

Partial Dependence (PD) and Individual Conditional Expectation (ICE): First suggested by [8], the PD is defined as the dependence of the prediction function

on x_S after all remaining features $X_{\setminus S}$ have been marginalized out [9]. The PD is estimated via Monte Carlo integration.

$$PD_{\hat{f},S}(x_S) = \mathbb{E}_{X_{\setminus S}}\left[\hat{f}(x_S, X_{\setminus S})\right] = \int \hat{f}(x_S, X_{\setminus S})\, d\mathcal{P}(X_{\setminus S}) \tag{1}$$

$$\widehat{PD}_{\hat{f},S}(x_S) = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_S, x_{\setminus S}^{(i)})$$

The PD is a useful feature effect measure when features are not interacting [8]. Otherwise it can obfuscate the relationships in the data [4]. In that case, the individual conditional expectation (ICE) can be used instead [9]. The i-th ICE corresponds to the expected value of the target for the i-th observation as a function of x_S, conditional on $x_{\setminus S}^{(i)}$.

$$\widehat{ICE}_{\hat{f},S}^{(i)}(x_S) = \hat{f}(x_S, x_{\setminus S}^{(i)})$$

The ICE disaggregates the global effect estimates of the PD to local effect estimates for single observations. Given $|S| = 1$, the ICE and PD are also referred to as ICE and PD curves. The ICE and PD suffer from extrapolation when features are correlated, because the permutations used to predict are located in regions without any training data [1].

Accumulated Local Effects (ALE): In [1] ALEs are presented as a feature effect measure for correlated features that does not extrapolate. The idea of ALEs is to take the integral with respect to X_j of the first derivative of the prediction function with respect to X_j. This creates an accumulated partial effect of X_j on the target variable while simultaneously removing additively linked effects of other features. The main advantage of not extrapolating stems from integrating with respect to the conditional distribution of $X_{\setminus j}$ on X_j instead of the marginal distribution of $X_{\setminus j}$ [1]. Let $z_{0,j}$ denote the minimum value of x_j. The first order ALE of the j-th feature at point x is defined as:

$$\begin{aligned} ALE_{\hat{f},j}(x) &= \int_{z_{0,j}}^{x} \mathbb{E}_{X_{\setminus j}|X_j}\left[\frac{\partial \hat{f}(X_j, X_{\setminus j})}{\partial X_j}\middle| X_j = z_j\right] dz_j - constant \\ &= \int_{z_{0,j}}^{x}\left[\int \frac{\partial \hat{f}(z_j, X_{\setminus j})}{\partial z_j}\, d\mathcal{P}(X_{\setminus j}|z_j)\right] dz_j - constant \end{aligned} \tag{2}$$

A constant is subtracted in order to center the plot. We estimate the first order ALE in three steps. First, we divide the value range of x_j into a set of intervals and compute a finite difference (FD) for each observation. For each i-th observation, $x_j^{(i)}$ is substituted by the corresponding right and left interval boundaries. Then the predictions with both substituted values are subtracted in order to receive an observation-wise FD. Second, we estimate local effects by averaging the FDs inside each interval. This replaces the inner integral in Eq. (2). Third, the accumulation of all local effects up to the point of interest replaces the outer integral in Eq. (2), i.e., the interval-wise average FDs are summed up.

The second order ALE is the bivariate extension of the first order ALE. It is important to note that first order effect estimates are subtracted from the second order estimates. In [1] the authors further lay out the computations necessary for higher order ALEs.

Marginal Effects (ME): MEs are an established technique in statistics and often used to interpret non-linear functions of coefficients in GLMs like logistic regression. The ME corresponds to the first derivative of the prediction function with respect to a feature at specified values of the input space. It is estimated by computing an observation-wise FD. The average marginal effect (AME) is the average of all MEs that were estimated with observed feature values [2]. Although there is extensive literature on MEs, this concept was suggested by [11] as a novel method for ML and referred to as the input gradient. Derivatives are also often utilized as a feature importance metric.

Shapley Value: Originating in coalitional game theory [19], the Shapley value is a local feature effect measure that is based on a set of desirable axioms. In coalitional games, a set of p players, denoted by P, play games and join coalitions. They are rewarded with a payout. The characteristic function $v : 2^p \to \mathbb{R}$ maps all player coalitions to their respective payouts [4]. The Shapley value is a player's average contribution to the payout, i.e., the marginal increase in payout for the coalition of players, averaged over all possible coalitions. For Shapley values as feature effects, predicting the target for a single observation corresponds to the game and a coalition of features represents the players. Shapley regression values were first developed for linear models with multicollinear features [13]. A model-agnostic Shapley value was first introduced in [19].

Consider the expected prediction for a single vector of feature values x, conditional on only knowing the values of features with indices in K ($K \subseteq P$), i.e., the features $X_{\setminus K}$ are marginalized out. This essentially equals a point (or a line, surface etc. depending on the power of K) on the PD from Eq. (1).

$$\mathbb{E}_{X_{\setminus K}}\left[\hat{f}(x_K, X_{\setminus K})\right] = \int \hat{f}(x_K, X_{\setminus K})\, d\mathcal{P}(X_{\setminus K}) = \widehat{PD}_{\hat{f},K}(x_K) \tag{3}$$

Equation (3) is shifted by the mean prediction and used as a payout function $v_{PD}(x_K)$, so that an empty set of features ($K = \varnothing$) results in a payout of zero [4].

$$\begin{aligned} v_{PD}(x_K) &= \mathbb{E}_{X_{\setminus K}}\left[\hat{f}(x_K, X_{\setminus K})\right] - \mathbb{E}_{X_{K\cup(P\setminus K)}}\left[\hat{f}(X_K, X_{\setminus K})\right] \\ &= \widehat{PD}_{\hat{f},K}(x_K) - \widehat{PD}_{\hat{f},\varnothing}(x_\varnothing) \\ &= \widehat{PD}_{\hat{f},K}(x_K) - \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_K^{(i)}, x_{\setminus K}^{(i)}) \end{aligned}$$

The marginal contribution $\Delta_j(x_K)$ of a feature value x_j joining the coalition of feature values x_K is:

$$\Delta_j(x_K) = v_{PD}(x_{K\cup\{j\}}) - v_{PD}(x_K) = \widehat{PD}_{\hat{f},K\cup\{j\}}(x_{K\cup\{j\}}) - \widehat{PD}_{\hat{f},K}(x_K)$$

The exact Shapley value of the j-th feature for a single vector of feature values x corresponds to:

$$\begin{aligned}\widehat{Shapley}_{\hat{f},j} &= \sum_{K \subseteq P \setminus \{j\}} \frac{|K|!(|P| - |K| - 1)!}{|P|!} \, \Delta_j(x_K) \\ &= \sum_{K \subseteq P \setminus \{j\}} \frac{|K|!(|P| - |K| - 1)!}{|P|!} \left[\widehat{PD}_{\hat{f},K \cup \{j\}}(x_{K \cup \{j\}}) - \widehat{PD}_{\hat{f},K}(x_K) \right]\end{aligned}$$

Shapley values are computationally expensive because the PD function has a complexity of $\mathcal{O}(N^2)$. Computations can be sped up by Monte Carlo sampling [19]. Furthermore, in [14] the authors propose a distinct variant to compute Shapley values called SHapley Additive exPlanations (SHAP).

Local Interpretable Model-Agnostic Explanations (LIME): In contrast to all previous techniques which are based on interpreting a single model, LIME [17] locally approximates the black box model with an intrinsically interpretable surrogate model. Given a single vector of feature values x, we first perturb x_j around a sufficiently close neighborhood while $x_{\setminus j}$ is kept constant. Then we predict with the perturbed feature values. The predictions are weighted by the proximity of the corresponding perturbed values to the original feature value. Finally, an intrinsically interpretable model is trained on the weighted predictions and interpreted instead.

4 Generalized Framework

Although the techniques presented in Sect. 3 are seemingly unrelated, they all work according to the exact same principle. Instead of trying to inspect the inner workings of a non-linear black box model, we evaluate its predictions when changing inputs. We can deconstruct model-agnostic techniques into a framework of four work stages: sampling, intervention, prediction, aggregation (SIPA). The software package `iml` [16] was inspired by the SIPA framework.

We first sample a subset (**sampling stage**) to reduce computational costs, e.g., we select a random set of available observations to evaluate as ICEs. In order to change the predictions made by the black box model, the data has to be manipulated. Feature values can be set to values from the observed marginal distributions (ICEs and PD or Shapley values), or to unobserved values (FD based methods such as MEs and ALEs). This crucial step is called the **intervention stage**. During the **prediction stage**, we predict on previously intervened data. This requires an already trained model, which is why model-agnostic techniques are always post-hoc. The predictions are further aggregated during the **aggregation stage**. Often, the predictions resulting from the prediction stage are local effect estimates, and the ones resulting from the aggregation stage are global effect estimates.

In Fig. 1, we demonstrate how all presented techniques for feature effects are based on the SIPA framework. Although LIME is a special case as it is based

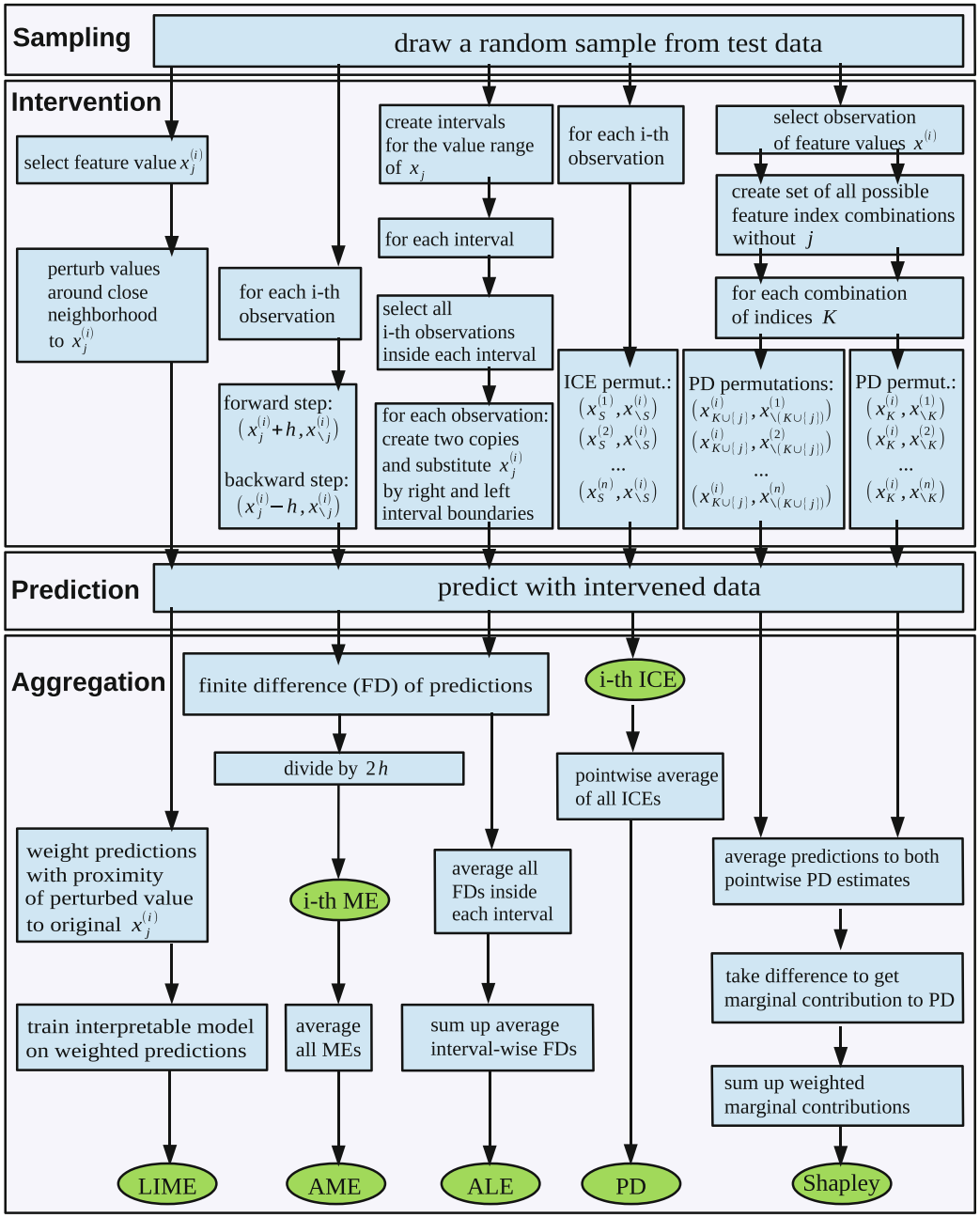


Fig. 1. We demonstrate how all presented model-agnostic methods for feature effects are based on the SIPA framework. For every method, we assign each computational step to the corresponding generalized SIPA work stage. Contrary to all other methods, LIME is based on training an intrinsically interpretable model during the aggregation stage. We consider training a model to be an aggregation, because it corresponds to an optimization problem where the training data is aggregated to a function. For reasons of simplicity, we do not differentiate between the actual functions or values and their estimates.

on training a local surrogate model, we argue that it is also based on the SIPA framework as training a surrogate model can be considered an aggregation of the training data to a function.

5 Feature Importance

We categorize model-agnostic importance measures into two groups: variance-based and performance-based.

Variance-Based: A mostly flat trajectory of a single ICE curve implies that in the underlying predictive model, varying x_j does not affect the prediction for this specific observation. If all ICE curves are shaped similarly, the PD can be used instead. In [10] the authors propose a measure for the curvature of the PD as a feature importance metric. Let the average value of the estimated PD of the j-th feature be denoted by $\widehat{\overline{PD}}_{\hat{f},j}(x_j) = \frac{1}{n}\sum_{i=1}^{n}\widehat{PD}_{\hat{f},j}(x_j^{(i)})$. The estimated importance $\widehat{\text{IMP}}_{\widehat{PD},j}$ of the j-th feature corresponds to the standard deviation of the feature's estimated PD function. The flatter the PD, the smaller its standard deviation and therefore the importance metric. For categorial features, the range of the PD is divided by 4. This is supposed to represent an approximation to the estimate of the standard deviation for small to medium sized samples [10].

$$\widehat{\text{IMP}}_{\widehat{PD},j} = \begin{cases} \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}\left[\widehat{PD}_{\hat{f},j}(x_j^{(i)}) - \widehat{\overline{PD}}_{\hat{f},j}(x_j)\right]^2} & x_j \text{ continuous} \\ \frac{1}{4}\left[max\left\{\widehat{PD}_{\hat{f},j}(x_j)\right\} - min\left\{\widehat{PD}_{\hat{f},j}(x_j)\right\}\right] & x_j \text{ categorial} \end{cases} \tag{4}$$

In [20] the authors propose the feature importance ranking measure (FIRM). They define a conditional expected score (CES) function for the j-th feature.

$$CES_{\hat{f},j}(v) = \mathbb{E}_{X_{\setminus j}}\left[\hat{f}(x_j, X_{\setminus j}) \mid x_j = v\right] \tag{5}$$

It turns out that Eq. (5) is equivalent to the PD from Eq. (1), conditional on $x_j = v$.

$$\begin{aligned} CES_{\hat{f},j}(v) &= \mathbb{E}_{X_{\setminus j}}\left[\hat{f}(v, X_{\setminus j})\right] \\ &= PD_{\hat{f},j}(v) \end{aligned}$$

The FIRM corresponds to the standard deviation of the CES function with all values of x_j used as conditional values. This in turn is equivalent to the standard deviation of the PD. The FIRM is therefore equivalent to the feature importance metric in Eq. (4).

$$\widehat{FIRM}_{\hat{f},j} = \sqrt{Var(\widehat{CES}_{\hat{f},j}(x_j))} = \sqrt{Var(\widehat{PD}_{\hat{f},j}(x_j))} = \widehat{\text{IMP}}_{\widehat{PD},j}$$

Performance-Based: The permutation feature importance (PFI), originally developed by [3] as a model-specific tool for random forests, was described as a model-agnostic one by [6]. If feature values are shuffled in isolation, the relationship between the feature and the target is broken up. If the feature is important for the predictive performance, the shuffling should result in an increased loss [4]. Permuting x_j corresponds to drawing from a new random variable $\tilde{X}_j$ that is distributed like X_j but independent of $X_{\setminus j}$ [4]. The model-agnostic PFI measures the difference between the generalization error (GE) on data with permuted and non-permuted values.

$$PFI_{\hat{f},j} = \mathbb{E}\left[\mathcal{L}(\hat{f}(\tilde{X}_j, X_{\setminus j}), Y)\right] - \mathbb{E}\left[\mathcal{L}(\hat{f}(X_j, X_{\setminus j}), Y)\right]$$

Let the permutation of x_j be denoted by $\tilde{x}_j$. Consider the sample of test data $\mathcal{D}_j$ where x_j has been permuted, and the non-permuted sample $\mathcal{D}$. The PFI estimate is given by the difference between GE estimates with permuted and non-permuted values.

$$\begin{aligned}\widehat{PFI}_{\hat{f},j} &= \widehat{GE}(\hat{f}, \mathcal{D}_j) - \widehat{GE}(\hat{f}, \mathcal{D}) \\ &= \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\hat{f}(\tilde{x}_j^{(i)}, x_{\setminus j}^{(i)}), y^{(i)}) - \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\hat{f}(x_j^{(i)}, x_{\setminus j}^{(i)}), y^{(i)})\end{aligned} \tag{6}$$

In [4] the authors propose individual conditional importance (ICI) and partial importance (PI) curves as visualization techniques that disaggregate the global PFI estimate. They are based on the same principle as the ICE and PD. The ICI visualizes the influence of a feature on the predictive performance for a single observation, while the PI visualizes the average influence of a feature for all observations. Consider the prediction for the i-th observation with observed values $\hat{f}(x_j^{(i)}, x_{\setminus j}^{(i)})$ and the prediction $\hat{f}(x_j^{(l)}, x_{\setminus j}^{(i)})$ where $x_j^{(i)}$ was replaced by a value $x_j^{(l)}$ from the marginal distribution of observed values x_j. The change in loss is given by:

$$\Delta\mathcal{L}^{(i)}(x_j^{(l)}) = \mathcal{L}(\hat{f}(x_j^{(l)}, x_{\setminus j}^{(i)})) - \mathcal{L}(\hat{f}(x_j^{(i)}, x_{\setminus j}^{(i)}))$$

The ICI curve of the i-th observation plots the value pairs $(x_j^{(l)}, \Delta\mathcal{L}^{(i)}(x_j^{(l)}))$ for all l values of x_j. The PI curve is the pointwise average of all ICI curves at all l values of x_j. It plots the value pairs $(x_j^{(l)}, \frac{1}{n}\sum_{i=1}^{n} \Delta\mathcal{L}^{(i)}(x_j^{(l)}))$ for all l values of x_j. Substituting values of x_j essentially resembles shuffling them. The authors demonstrate how averaging the values of the PI curve results in an estimation of the global PFI.

$$\widehat{PFI}_{\hat{f},j} = \frac{1}{n}\sum_{l=1}^{n}\frac{1}{n}\sum_{i=1}^{n} \Delta\mathcal{L}^{(i)}(x_j^{(l)})$$

Furthermore, a feature importance measure called Shapley feature importance (SFIMP) was proposed in [4]. Shapley importance values based on model

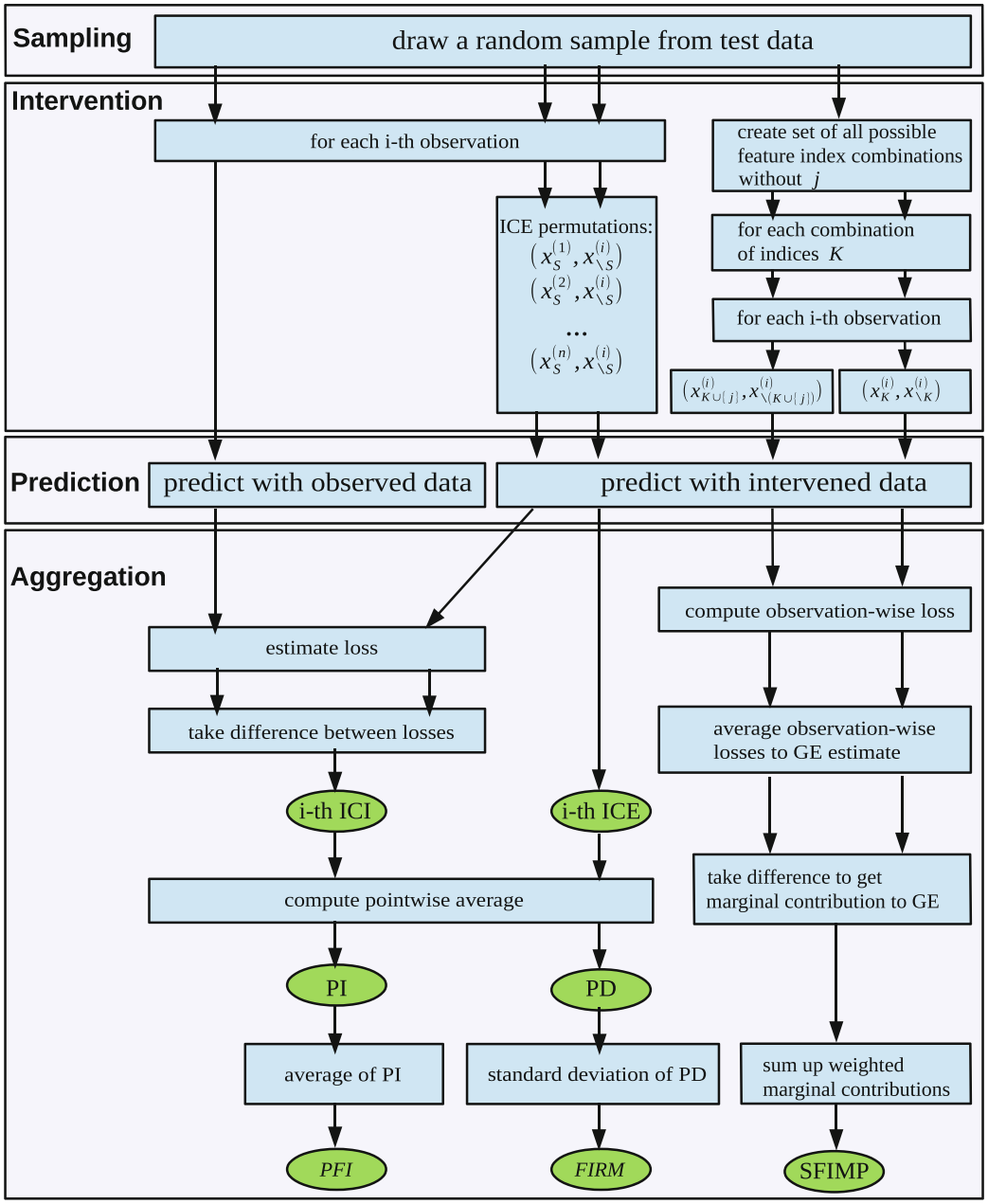


Fig. 2. We demonstrate how importance computations are based on the same work stages as effect computations. In the same way as in Fig. 1, we assign the computational steps of all techniques to the corresponding generalized SIPA work stages. Variance-based importance measures such as FIRM measure the variance of a feature effect, i.e., we add a variance computation during the aggregation stage. Performance-based importance measures such as ICI, PI, PFI and SFIMP are based on computing changes in loss after the intervention stage. For reasons of simplicity, we do not differentiate between the actual functions or values and their estimates.

refits with distinct sets of features were first introduced by [5] for feature selection. This changes the behavior of the learning algorithm and is not helpful to evaluate a single model, as noted by [4]. The SFIMP is based on the same computations as the Shapley value but replaces the payout function with one that is sensitive to the model performance. The authors define a new payout $v_{GE}(x_j)$ that substitutes the estimated PD with the estimated GE. This is equivalent to the estimated PFI from Eq. (6).

$$v_{GE}(x_j) = \widehat{GE}(\hat{f}, \mathcal{D}_j) - \widehat{GE}(\hat{f}, \mathcal{D}) = \widehat{PFI}_{\hat{f},j} = v_{PFI}(x_j)$$

We can therefore refer to $v_{GE}(x_j)$ as $v_{PFI}(x_j)$ and regard the SFIMP as an extension to the PFI [4].

6 Extending the Framework to Importance Computations

Variance-based importance methods measure the variance of feature effect estimates, which we already demonstrated to be based on the SIPA framework. Therefore, we simply add a variance computation during the aggregation stage. Performance-based techniques measure changes in loss, i.e., there are two possible modifications. First, we predict on non-intervened or intervened data (prediction stage). Second, we aggregate predictions to the loss (aggregation stage). In Fig. 2, we demonstrate how feature importance computations are based on the same work stages as feature effect computations.

7 Conclusion

In recent years, various model-agnostic interpretation methods have been developed. Due to different notations and terminology it is difficult to see how they are related. By deconstructing them into sequential work stages, one discovers striking similarities in their methodologies. We first provided a survey on model-agnostic interpretation methods and then presented the generalized SIPA framework of sequential work stages. First, there is a sampling stage to reduce computational costs. Second, we intervene in the data in order to change the predictions made by the black box model. Third, we predict on intervened or non-intervened data. Fourth, we aggregate the predictions. We embedded multiple methods to estimate the effect (ICE and PD, ALEs, MEs, Shapley values and LIME) and importance (FIRM, PFI, ICI and PI and the SFIMP) of features into the framework. By pointing out how all demonstrated techniques are based on a single methodology, we hope to work towards a more unified view on model-agnostic interpretations and to establish a common ground to discuss them in future work.

Acknowledgments. This work is supported by the Bavarian State Ministry of Science and the Arts as part of the Centre Digitisation.Bavaria (ZD.B) and by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibilities for its content.

References

1. Apley, D.W.: Visualizing the effects of predictor variables in black box supervised learning models. arXiv e-prints arXiv:1612.08468, December 2016
2. Bartus, T.: Estimation of marginal effects using margeff. Stata J. **5**(3), 309–329 (2005)
3. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
4. 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
5. Cohen, S., Dror, G., Ruppin, E.: Feature selection via coalitional game theory. Neural Comput. **19**(7), 1939–1961 (2007)
6. Fisher, A., Rudin, C., Dominici, F.: Model class reliance: variable importance measures for any machine learning model class, from the "Rashomon" perspective. arXiv e-prints arXiv:1801.01489, January 2018
7. Fisher, A., Rudin, C., Dominici, F.: All models are wrong but many are useful: variable importance for black-box, proprietary, or misspecified prediction models, using model class reliance. arXiv e-prints arXiv:1801.01489, January 2018
8. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**(5), 1189–1232 (2001)
9. 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**, 44–65 (2013)
10. Greenwell, B.M., Boehmke, B.C., McCarthy, A.J.: A simple and effective model-based variable importance measure. arXiv e-prints arXiv:1805.04755, May 2018
11. Hechtlinger, Y.: Interpretation of prediction models using the input gradient. arXiv e-prints arXiv:1611.07634, November 2016
12. Leeper, T.J.: Margins: marginal effects for model objects (2018)
13. Lipovetsky, S., Conklin, M.: Analysis of regression in game theory approach. Appl. Stoch. Models Bus. Ind. **17**(4), 319–330 (2001)
14. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems, vol. 30, pp. 4765–4774. Curran Associates, Inc., New York (2017)
15. Molnar, C.: Interpretable Machine Learning (2019). https://christophm.github.io/interpretable-ml-book/
16. Molnar, C., Bischl, B., Casalicchio, G.: iml: an R package for interpretable machine learning. JOSS **3**(26), 786 (2018)
17. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should I trust you?: explaining the predictions of any classifier. In: Knowledge Discovery and Data Mining (KDD) (2016)
18. Rudin, C., Ertekin, Ş.: Learning customized and optimized lists of rules with mathematical programming. Math. Program. Comput. **10**(4), 659–702 (2018). https://doi.org/10.1007/s12532-018-0143-8
19. 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
20. Zien, A., Krämer, N., Sonnenburg, S., Rätsch, G.: The feature importance ranking measure. In: Buntine, W., Grobelnik, M., Mladenić, D., Shawe-Taylor, J. (eds.) ECML PKDD 2009. LNCS (LNAI), vol. 5782, pp. 694–709. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-04174-7_45

6 General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models

Contributing Paper

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: *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*. Ed. by Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.-R., and Samek, W. Vol. 13200. Lecture Notes in Computer Science. Cham: Springer International Publishing, pp. 39–68. DOI: 10.1007/978-3-031-04083-2_4

Declaration of Contributions

C.A. Scholbeck contributed to the conceptualization of the project, wrote parts of the introduction, and assisted in revising the paper.

C. Molnar initiated and coordinated the project. C. Molnar, G. König, J. Herbinger, T. Freiesleben, S. Dandl, and G. Casalicchio co-authored at least one chapter. All authors assisted in revising 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

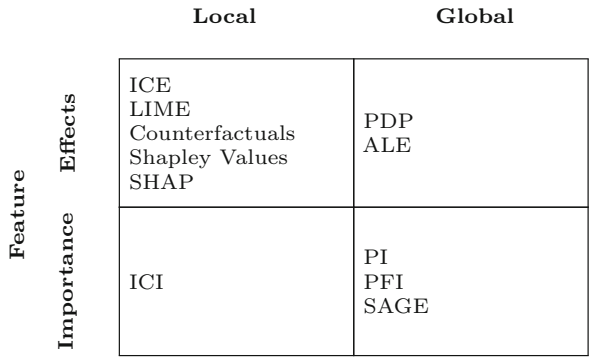


Fig. 1. Selection of popular model-agnostic interpretation techniques, classified as local or global, and as effect or importance methods.

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

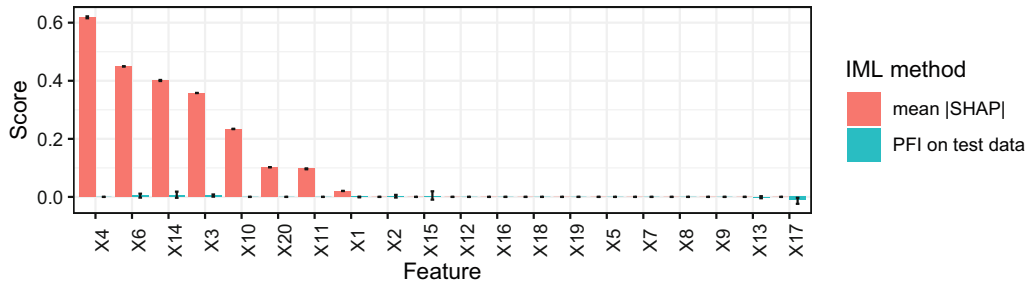


Fig. 2. Assuming one-fits-all interpretability. A default `xgboost` regression model that minimizes the mean squared error (MSE) was fitted on 20 independently and uniformly distributed features to predict another independent, uniformly sampled target. In this setting, predicting the (unconditional) 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

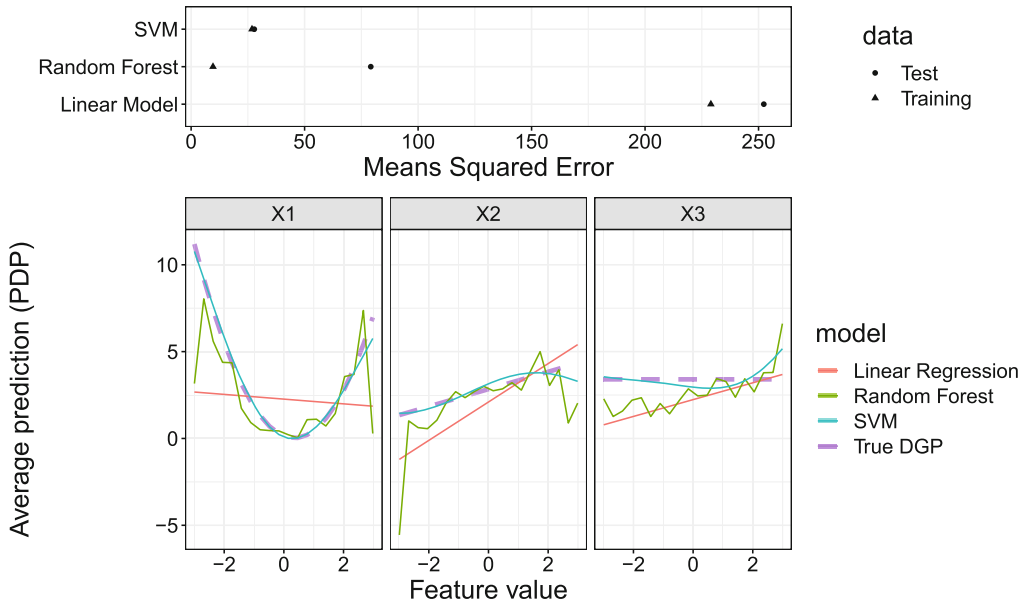


Fig. 3. Bad model generalization. **Top:** Performance estimates on training and test data for a linear regression model (underfitting), a random forest (overfitting) and a support vector machine with radial basis kernel (good fit). The three features are drawn from a uniform distribution, and the target 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

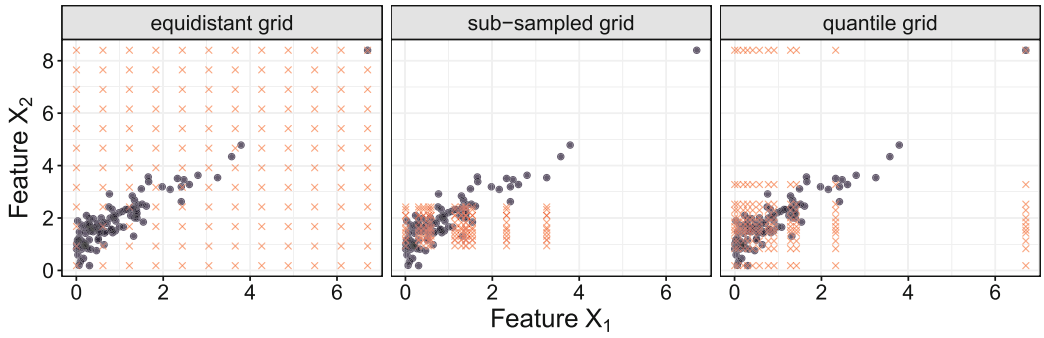


Fig. 4. Interpretation with extrapolation. Illustration of artificial data points generated by three different perturbation approaches. The black dots refer to observed data points and the red crosses to the artificial data points.

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].

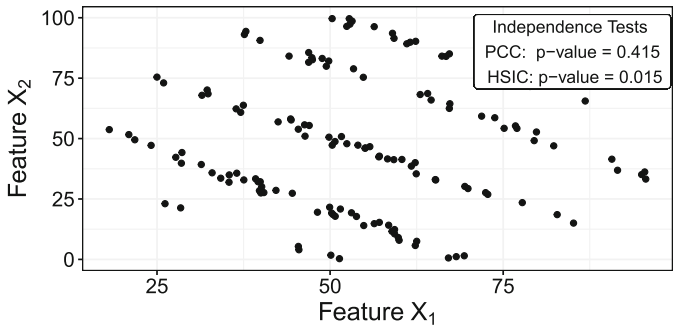


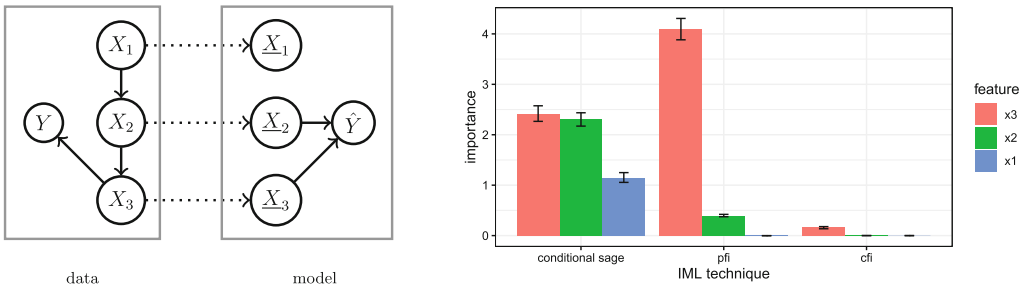
Fig. 5. Confusing linear correlation with dependence. Highly dependent features X_1 and X_2 that have a correlation close to zero. A test (H_0: Features are independent) using Pearson correlation is not significant, but for HSIC, the H_0-hypothesis gets rejected. Data from [80].

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.

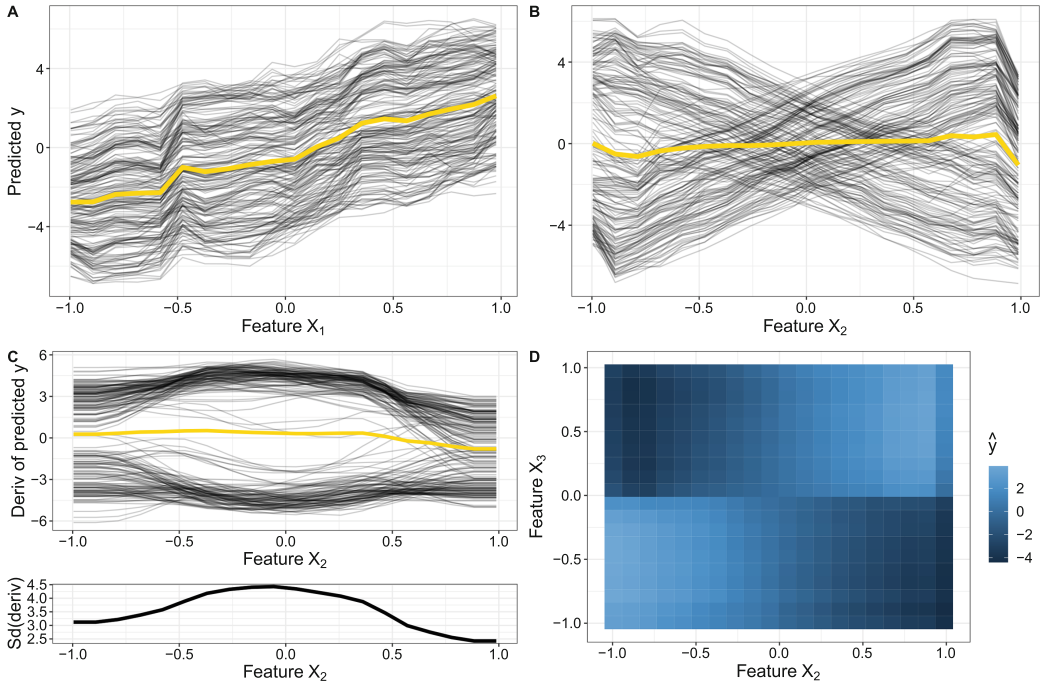


Fig. 7. Misleading effect due to interactions. Simulation example with interactions: $Y = 3X_1 - 6X_2 + 12X_2\mathbb{1}_{(X_3 \geq 0)} + \epsilon$ with $X_1, X_2, X_3 \overset{i.i.d.}{\sim} U[-1, 1]$ and $\epsilon \overset{i.i.d.}{\sim} N(0, 0.3)$. A random forest with 500 trees is fitted on 1000 observations. 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.

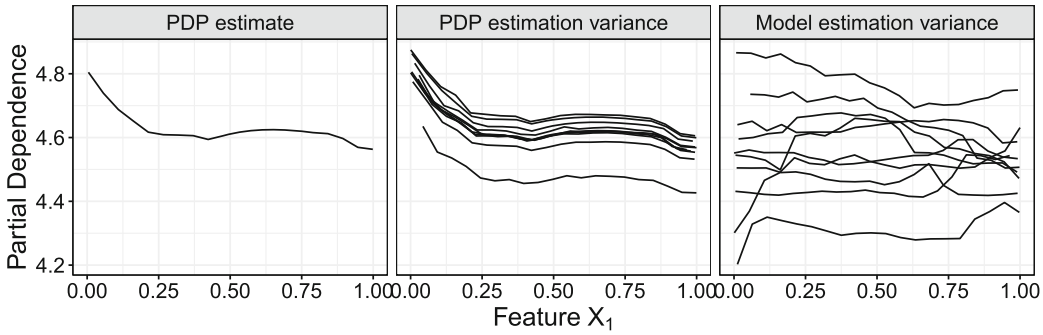


Fig. 8. Ignoring model and approximation uncertainty. PDP for X_1 with $Y = 0 \cdot X_1 + \sum_{j=2}^{10} X_j + \epsilon_i$ with $X_1, \ldots, X_{10} \sim U[0,1]$ and $\epsilon_i \sim N(0, 0.9)$. **Left:** PDP for X_1 of a random forest trained on 100 data points. **Middle:** Multiple PDPs (10x) for 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.

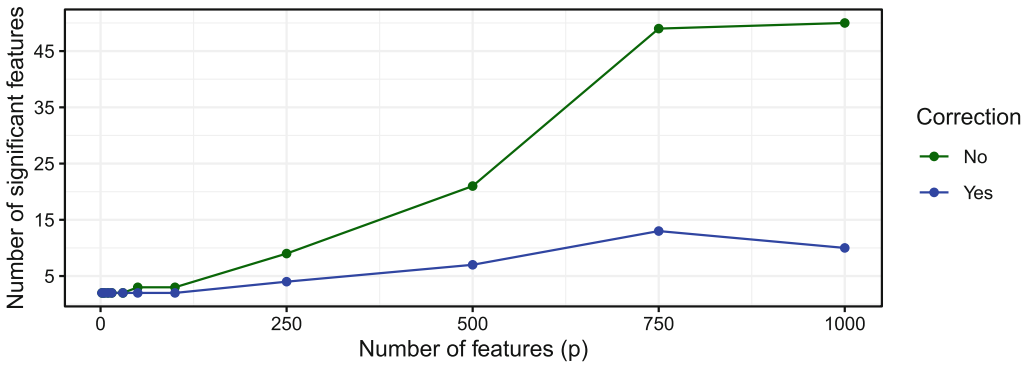


Fig. 9. Failure to scale to high-dimensional settings. Comparison of the number of features with significant importance - once with and once without Bonferroni-corrected significance levels for a varying number of added noise variables. Datasets were sampled from $Y = 2X_1 + 2X_2^2 + \epsilon$ with $X_1, X_2, \epsilon \sim N(0,1)$. $X_3, X_4, ..., X_p \sim N(0,1)$ 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.

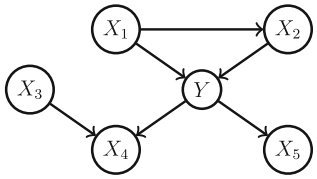


Fig. 10. Causal graph

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 ρ. 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

7 Position Paper: Bridging the Gap Between Machine Learning and Sensitivity Analysis

Contributing Paper

Scholbeck, C. A., Moosbauer, J., Casalicchio, G., Gupta, H., Bischl, B., and Heumann, C. (2023b). "Position Paper: Bridging the Gap Between Machine Learning and Sensitivity Analysis". In: arXiv: 2312.13234 [cs.LG]

Declaration of Contributions

C.A. Scholbeck contributed to this paper as the first author. C.A. Scholbeck conceptualized the project, reviewed the literature, drafted the paper, and revised it according to the feedback from his co-authors.

J. Moosbauer contributed to the section introducing ML and HPO. J. Moosbauer, G. Casalicchio, H. Gupta, B. Bischl, and C. Heumann provided valuable input and assisted in revising the paper.

Position Paper: Bridging the Gap Between Machine Learning and Sensitivity Analysis

Christian A. Scholbeck [1 2 3] **Julia Moosbauer** [1 2] **Giuseppe Casalicchio** [1 2] **Hoshin Gupta** [3] **Bernd Bischl** [1 2]
Christian Heumann [1]

Abstract

We argue that interpretations of machine learning (ML) models or the model-building process can be seen as a form of sensitivity analysis (SA), a general methodology used to explain complex systems in many fields such as environmental modeling, engineering, or economics. We address both researchers and practitioners, calling attention to the benefits of a unified SA-based view of explanations in ML and the necessity to fully credit related work. We bridge the gap between both fields by formally describing how (a) the ML process is a system suitable for SA, (b) how existing ML interpretation methods relate to this perspective, and (c) how other SA techniques could be applied to ML.

1. Introduction

Machine learning (ML) is concerned with learning models from data with applications as diverse as text (Zhang et al., 2015) and speech processing (Bhangale & Mohanaprasad, 2021), robotics (Pierson & Gashler, 2017), medicine (Rajkomar et al., 2019), climate research (Rolnick et al., 2022), or finance (Huang et al., 2020). Due to the increasing availability of data and computational resources, demand for ML has risen sharply in recent years, permeating all aspects of life. While the first publications in predictive modeling date back as far as the 1800s with Gauß and Legendre (Molnar et al., 2020; Stigler, 1981), the popularity of ML has surged in the twenty-first century, as it represents the current technological backbone for artificial intelligence. Increasing focus is put on interpretable models or the interpretation of black box models with model-agnostic techniques (Molnar, 2022; Rudin et al., 2022), often referred to as interpretable ML (IML) or explainable artificial intelligence. Note that we utilize the term black box, although the internal workings of a model may be accessible but too complex for the human mind to comprehend. Furthermore, interpretations of the hyperparameter optimization (HPO) process have garnered attention in recent years (Hutter et al., 2014). In the context of this paper, we will refer to IML as any effort to gain an understanding of ML, including HPO.

In a basic sense, sensitivity analysis (SA) (Saltelli et al., 2008; Razavi et al., 2021; Iooss & Lemaître, 2015) is the study of how model output is influenced by model inputs. It is used as an assistance in many fields to explain input-output relationships of complex systems. Applications include environmental modeling (Song et al., 2015; Wagener & Pianosi, 2019; Shin et al., 2013; Haghnegahdar & Razavi, 2017; Gao et al., 2023; Mai et al., 2022; Nossent et al., 2011), biology (Sumner et al., 2012), engineering (Guo et al., 2016; Ballester-Ripoll et al., 2019; Becker et al., 2011), nuclear safety (Saltelli & Tarantola, 2002), energy management (Tian, 2013), economics (Harenberg et al., 2019; Ratto, 2008), or financial risk management (Baur et al., 2004). In some jurisdictions such as the European Union, SA is officially required for policy assessment (Saltelli et al., 2019). With roots in design of experiments (DOE), SA started to materialize in the 1970s and 1980s with the availability of computational resources and the extension of DOE to design of computer experiments (DACE); its large body of research is however spread across various disciplines, resulting in a lack of visibility (Razavi et al., 2021).

Why This Position Paper: ML evolved largely independent of SA. As a result, the community did not fully credit related work and left potential research gaps unexplored. For instance, the high-dimensional model representation (HDMR) dates back to Hoeffding (1948) and is the basis for variance-based SA, including Sobol indices (Sobol, 1990; Homma & Saltelli, 1996; Rabitz & Aliş, 1999; Saltelli et al., 2008). For the HDMR, decomposing the function into lower-dimensional terms is instrumental, an approach which was later redeveloped for ML and termed partial dependence (PD) (Friedman, 2001). Furthermore, the HDMR is now better known as the functional analysis of variance (FANOVA)

[1]Department of Statistics, Ludwig-Maximilians-Universität in Munich, Munich, Germany [2]Munich Center for Machine Learning (MCML), Munich, Germany [3]Department of Hydrology and Atmospheric Sciences, The University of Arizona, Tucson AZ, USA. Correspondence to: Christian A. Scholbeck <christian.scholbeck@lmu.de>.

decomposition in ML (Hooker, 2004; 2007; Molnar, 2022) while many people are unaware of its roots in SA. Both the FANOVA (Hutter et al., 2014) and the PD (Moosbauer et al., 2021) have also been used to explain the HPO process. In algorithm configuration problems such as for HPO, evaluating paths of configurations by iteratively modifying parameters is known as ablation analysis (Fawcett & Hoos, 2016; Biedenkapp et al., 2017) but is strikingly similar to existing work in SA such as one-factor-at-a-time methods (Saltelli et al., 2008). Recent advances in Shapley values (Štrumbelj & Kononenko, 2014) have been made both in the SA (Owen, 2014) as well as the ML community (Lundberg & Lee, 2017). Several techniques have been developed to determine the importance of features in ML (Hooker et al., 2021; Casalicchio et al., 2019; Fisher et al., 2019; Molnar et al., 2024), but numerous advances in other fields to compute variable importances (Wei et al., 2015) are often overlooked. Grouping features for importance computations has only recently become relevant in ML (Au et al., 2022), although grouping system inputs has been an important topic in SA for decades (Sheikholeslami et al., 2019a).

Our Position: We argue that IML can be seen as a form of SA applied to ML, which integrates recent advances in IML into a larger body of research on how to explain complex systems. We call attention to the benefits of a unified SA-based view of explanations in ML, to the necessity to fully credit related work, and to potential research gaps.

To substantiate our claim, we bridge the gap between ML and SA by (a) formally describing the ML process as a system suitable for SA, (b) highlighting how existing methods relate to this perspective, and (c) discussing how SA methods—which are typically used in the domain sciences—can be applied to ML.

2. Related Work

In their survey of methods to explain black box models, Guidotti et al. (2018) mention SA as one approach besides other explanation methods, a distinction we aim to abandon in the context of this paper. Razavi et al. (2021) revisit the status of SA and conceive a vision for its future, including connections to ML regarding feature selection, model interpretations, and ML-powered SA; although they provide several clues about the connections between ML and SA, the paper does not formalize the interpretation process in ML and how it relates to SA in detail. Scholbeck et al. (2020) present a generalized framework of work stages for model-agnostic interpretation methods in ML, consisting of a sampling, intervention, prediction, and aggregation stage; although this methodology resembles SA (in the sense that an intervention in feature values is followed by a prediction with the trained model), the paper does not establish a formal connection. Several authors applied traditional SA methods to ML: Fel et al. (2021) describe the importance of regions in image data with Sobol indices; Kuhnt & Kalka (2022) provide a short overview on variance-based SA for ML model interpretations; Stein et al. (2022) provide a survey of SA methods for model interpretations, conduct an analysis under different conditions, and apply the Morris method to compute sensitivity indices for a genomic prediction task; Paleari et al. (2021) use the Morris method to rank the feature importance for a generic crop model; Tunkiel et al. (2020) use derivative-based SA to rank high-dimensional features for a directional drilling model; Ojha et al. (2022) evaluate the sensitivity of the model performance regarding hyperparameters using the Morris method and Sobol indices.

Many authors have given considerable thought to the connection between ML and SA, either in the sense that SA can be applied to ML and vice versa or that methods in IML resemble some form of SA. We see this work here as connecting the dots between these efforts, establishing a formal link between ML and SA, directing attention to overlooked related work, and as a result, bringing both communities together.

3. An Introduction to ML and SA

3.1. Supervised Machine Learning

In supervised learning, a model is learned from labeled data to predict based on new data from the same distribution with minimal error. To be precise, supervised learning requires a labeled data set $\mathcal{D} = \left(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\right)_{i=1,\ldots,n}$ of observations $\left(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\right)$ where $\boldsymbol{x}^{(i)}$ corresponds to the p-dimensional feature vector drawn from the feature space $\mathcal{X}$ and $\boldsymbol{y}^{(i)}$ to the g-dimensional target vector (also referred to as label) drawn from the target space $\mathcal{Y}$. We assume observations are drawn i.i.d. from an unknown distribution $\mathbb{P}_{\boldsymbol{xy}}$ which is specific to the underlying learning problem:

$$\left(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\right) \sim \mathbb{P}_{\boldsymbol{xy}}$$

We formalize the concept of training by introducing an inducer (or learner) $\mathcal{I}$ as a function that maps the training subset $\mathcal{D}_{\text{train}} \subset \mathcal{D}$ with n_{train} observations and hyperparameter configuration $\boldsymbol{\lambda} \in \Lambda$ to a model $\widehat{f}$ from a hypothesis space $\mathcal{H}$:

$$\mathcal{I} : \begin{cases} \mathcal{X} \times \mathcal{Y} \times \Lambda & \to \mathcal{H} \\ (\mathcal{D}_{\text{train}}, \boldsymbol{\lambda}) & \mapsto \widehat{f} \end{cases}$$

Many learners use empirical risk minimization to train $\widehat{f}$:

$$R_{\text{emp}}(\tilde{f}) = \frac{1}{n_{\text{train}}} \sum_{i \,:\, (\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}) \in \mathcal{D}_{\text{train}}} L(\tilde{f}(\boldsymbol{x}^{(i)}), \boldsymbol{y}^{(i)})$$

$$\widehat{f} = \arg\min_{\tilde{f}} R_{\text{emp}}(\tilde{f})$$

R_{emp} is only a proxy for the true generalization error (GE):

$$\text{GE} = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathbb{P}_{\boldsymbol{x},\boldsymbol{y}}}\left[L(\widehat{f}(\boldsymbol{x}),\boldsymbol{y})\right]$$

$\widehat{f}$ is finally evaluated on an outer performance measure ρ (which may coincide with L) on a test subset $\mathcal{D}_{\text{test}} \subset \mathcal{D}$:

$$\rho : \begin{cases} \mathcal{X}\times\mathcal{Y}\times\mathcal{H}\to\mathbb{R} \\ (\mathcal{D}_{\text{test}},\widehat{f}) \mapsto \widehat{\text{GE}} \end{cases}$$

Experience has shown that resampling (and aggregating results) is a more efficient use of data; $\mathcal{D}$ can be repeatedly split up into different train and test sets; $\mathcal{D}_{\text{train}}$ can be further split up in an inner loop to optimize hyperparameters (Bischl et al., 2023).

3.1.1. Hyperparameter Optimization

In addition to controlling the behavior of the learner $\mathcal{I}$, the entire learning procedure is configurable by $\boldsymbol{\lambda}$ which may, for example, control the hypothesis space (e.g., the number of layers of a neural network), the training process (e.g., a learning rate, regularization parameter, or resampling splits), or the data (e.g., a subsampling rate).

We formalize the input-output relationship between $\boldsymbol{\lambda}$ and the GE estimate as the function c:

$$c : \begin{cases} \Lambda & \to \mathbb{R} \\ \boldsymbol{\lambda} & \mapsto \widehat{GE} \end{cases}$$

For most learning problems, this relationship is non-trivial, and it is of concern to select the hyperparameter configuration carefully. This has given rise to the HPO problem:

$$\boldsymbol{\lambda}^* \in \arg\min_{\boldsymbol{\lambda}\in\Lambda} c(\boldsymbol{\lambda})$$

In practice, one is typically interested in finding a configuration $\widehat{\boldsymbol{\lambda}}$ with a performance close to the theoretical optimum $c(\widehat{\boldsymbol{\lambda}}) \approx c(\boldsymbol{\lambda}^*)$. Solving the optimization problem is challenging: usually, no analytical information about the objective function c is available, evaluations of c are expensive (a single evaluation of c requires executing a training run), and the hyperparameter space might have a complex structure (hierarchical, mixed numeric-categorical).

Even experts find manual hyperparameter tuning through trial-and-error to be challenging and time-consuming, which creates a demand for HPO algorithms capable of efficiently discovering good solutions. Grid search (evaluating c on an equidistant set of grid points in Λ) and random search (evaluating c on a set of randomly sampled points in Λ) are the simplest approaches. More recently, Bayesian optimization (Jones et al., 1998) has become increasingly popular for hyperparameter tuning (Hutter et al., 2011; Snoek et al., 2012).

Without going into too much detail on the vast number of approaches in HPO (Bischl et al., 2023; Feurer & Hutter, 2019), we simply denote a generic hyperparameter tuner by:

$$\tau : (\mathcal{D},\mathcal{I},\Lambda,L,\rho) \mapsto \widehat{\boldsymbol{\lambda}}$$

It maps the data $\mathcal{D}$, inducer $\mathcal{I}$, hyperparameter search space Λ, inner loss function L, and outer performance measure ρ to the estimated best hyperparameter configuration $\widehat{\boldsymbol{\lambda}}$.

Automated ML and Meta-Learning: The quality of an ML model is sensitive to many steps that need to be performed before actual training, including cleaning and preparation of a data set, feature selection or feature engineering, dimensionality reduction, selecting a suitable model class, and performing HPO. This process can be illustrated as a pipeline of work stages. The space of possible pipeline configurations can be seen as a (typically mixed-hierarchical) hyperparameter space, which HPO algorithms can optimize over, and which is the subject of automated ML.

Also described as "learning to learn", meta-learning is concerned with the study of how ML techniques perform on different tasks and using this knowledge to build models faster and with better performance (Vanschoren, 2018). In particular, one is interested in learning how task characteristics influence the behavior of learners. Meta-learning can be considered an abstraction of HPO where the task characteristics represent hyperparameters.

3.1.2. Interpretable Machine Learning

Model Interpretations: In recent years, explanations of the relationship between features and model predictions or features and the model performance have become an important part of ML (Molnar, 2022). There is no consensus regarding the definition or quantification of model interpretability, e.g., it can comprise sparsity of the model, the possibility of visual interpretations, decomposability into sub-models, and many other characteristics (Rudin et al., 2022). Some model types can be interpreted based on model-specific characteristics (also referred to as intrinsically interpretable models), e.g., (generalized) linear models, generalized additive models, or decision trees. But many ML models, including random forests, gradient boosting, support vector machines, or neural networks, generally are black boxes. We can gain insights into the workings of such black boxes with model-agnostic techniques, which are applicable to any model type.

A general explanation process can be formalized as a function Γ that maps the model $\widehat{f}$, explanation parameter configuration $\boldsymbol{\eta}$, and a data set $\mathcal{D}_{\text{explain}}$ to an explanation Ξ:

$$\Gamma : (\widehat{f},\boldsymbol{\eta},\mathcal{D}_{\text{explain}}) \mapsto \Xi$$

$\mathcal{D}_{\text{explain}}$ can be used to query the model and may consist of training, test, or artificial feature values, as well as observed

target values for loss-based methods (Fisher et al., 2019; Casalicchio et al., 2019; Scholbeck et al., 2020). Depending on the explanation method, $\boldsymbol{\eta}$ may, for instance, control how to train a surrogate model or how to select or manipulate values in $\mathcal{D}_{\text{explain}}$.

Ξ can be a scalar value: for instance, the permutation feature importance (PFI) (Fisher et al., 2019) and H-statistic (Friedman & Popescu, 2008) indicate the importance of features or the interaction strength between features, respectively.

Ξ may also consist of a set of values indicating the effect of a subset of features on the predicted outcome, e.g., the individual conditional expectation (ICE) (Goldstein et al., 2015), partial dependence (PD) (Friedman, 2001), accumulated local effects (ALE) (Apley & Zhu, 2020), or Shapley values (Štrumbelj & Kononenko, 2014; Lundberg & Lee, 2017). Some methods adapt this methodology to evaluate the prediction loss (Casalicchio et al., 2019). Such functions, conditional on being lower-dimensional, also serve as visualization tools.

Ξ can consist of data points for methods such as counterfactual explanations (Wachter et al., 2018; Dandl et al., 2020), which search for the smallest necessary changes in feature values to receive a targeted prediction.

Furthermore, some methods replace a complex non-interpretable model with a less complex interpretable one; here, Ξ is a set of predictions returned by the surrogate. We differentiate between global surrogates, that train a surrogate on the entire feature space, or local ones, which do so for a single data point, e.g., local interpretable model-agnostic explanations (LIME) (Ribeiro et al., 2016).

Note that Γ always provides information about how predictions of the trained model $\widehat{f}$ are influenced by the features. A detailed illustration of this process for many model-agnostic techniques (which query the model with different feature values) such as the PFI, ICE, PD, ALE, LIME, or Shapley values is provided by Scholbeck et al. (2020). The same holds, in general, for model-specific interpretations, which provide similar insight without querying the model. For instance, in linear regression models, beta coefficients are equivalent to certain sensitivity indices based on model queries (Saltelli et al., 2008).

HPO Explanations: A distinct branch of IML is concerned with explanations of the HPO process (formalized by c). Hutter et al. (2014) compute a FANOVA of c with a random forest surrogate model. Moosbauer et al. (2021) compute a PD of c with uncertainty estimate enhancements. An ablation analysis (Fawcett & Hoos, 2016; Biedenkapp et al., 2017) can be used to evaluate effects of iterative modifications of hyperparameters on the performance. Woźnica & Biecek (2021) explore the interpretation of meta models.

3.2. Sensitivity Analysis

3.2.1. Systems Modeling

SA is concerned with modeling systems that consist of one or multiple interconnected models or functions (Razavi et al., 2021) A model ϕ can either be determined manually or data-driven. The former is also referred to as law-driven, mechanistic, or process-based. A model receives an input vector $\boldsymbol{z} = (z_1, \ldots, z_l) \in \mathbb{R}^l$ and returns an output vector $\boldsymbol{q} = (q_1, \ldots, q_v) \in \mathbb{R}^v$:

$$\phi(\boldsymbol{z}) = \boldsymbol{q}$$

As an illustration, consider a simple system that consists of two models where a model ϕ_2 receives the scalar output of another model ϕ_1 as an input:

$$\begin{aligned} \phi_1(z_1) &= q_1 \\ \phi_2(q_1) &= q_2 \end{aligned} \tag{1}$$

Systems are typically illustrated visually. As an example, consider the HBV-SASK hydrological system (Gupta & Razavi, 2018). One option is to visualize each model as a node that is connected to other nodes by streams of inputs and outputs (see Fig. 1). By building up a system of many

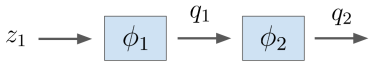


Figure 1. One option to visualize the example system in Eq. (1).

smaller models instead of creating a single large model (quasi, creating a "network of models"), systems modeling allows for more sophisticated relationships between components while reducing complexity. As an example, consider an earth system model which could be composed of components modeling the atmosphere, ocean, land, and sea ice, and the exchange of energy and mass between these parts (Heavens et al., 2013). Whereas for a conventional singular climate model, one would require input data on carbon dioxide, earth system models can directly use anthropogenic emissions (caused by humans) and express the link between policies and climate change more explicitly (Kawamiya et al., 2020).

3.2.2. SA of a System

We analyze such a system for various purposes: assessing the similarity between a model and the underlying real phenomenon, determining the importance of input factors, identifying regions of the input space that contribute the most to output variability, evaluating the interdependence between input factors in their influence on the output, or identifying non-influential factors for the purpose of model simplification (Razavi & Gupta, 2015).

An input within the system (which we then manipulate for our analysis) can be any variable factor, e.g., model input values, model parameters, or constraints. We differentiate between local SA (LSA), which investigates a single location in the input space, and global SA (GSA), where input factors are varied simultaneously. In LSA, the inputs are deterministic, while in GSA, $\boldsymbol{z}$ and $\boldsymbol{q}$ are considered random vectors with probability distributions (Borgonovo & Plischke, 2016). LSA does not capture the aggregate behavior of a model, i.e., it can only provide insights into the influence of an input factor on the output for a single configuration (i.e., all remaining factors are kept constant). GSA has been developed with the motivation of explaining the model behavior across the entire input space. However, as opposed to local sensitivity, there is no unique definition of GSA methods (Razavi & Gupta, 2015), which vary considerably in terms of their methodological approach (Iooss & Lemaître, 2015; Borgonovo & Plischke, 2016; Razavi et al., 2021).

The output of an SA method does not have to be identical to the output of a model. For instance, Monte-Carlo filtering produces random model outputs and identifies the ones that are located inside a region of interest (often referred to as the behavioral region) and the ones in the remaining regions (referred to as non-behavioral regions) (Saltelli et al., 2008). Monte-Carlo filtering is used in regional SA which aims to identify the most important input factors that lead to model outputs in the behavioral region. A "setting" determines how a factor's relevance or importance is defined and how it should be investigated, thereby justifying the use of certain methods for the task at hand. Common settings include factor prioritization and factor fixing, which aim to determine the most and least important input factors, respectively. The former is suited to rank the importance of input factors, while the latter is suited for screening input factors.

3.2.3. SA Methods

SA methods can be categorized in multiple ways. This section provides an exemplary and non-exhaustive overview. The interested reader may be referred to the works of Saltelli et al. (2008), Borgonovo & Plischke (2016), Pianosi et al. (2016), or Razavi et al. (2021) for further insights (or slightly different categorizations).

Finite-difference-based methods aggregate finite differences (FDs) gathered at various points of the input space for a global representation of input influence. Input perturbations range from very small (numeric derivatives) to larger magnitudes. The Morris method, also referred to as the elementary effects (EE) method (Morris, 1991), creates paths through the input space by traversing it one factor at a time, evaluating the model at each step of the path. Due to its low computational cost, the EE method is an important screening technique in SA to this date and has been modified numerous times (Saltelli et al., 2008; Campolongo et al., 2007). One-factor-at-a-time methods are often criticized for leaving important areas of the feature space unexplored (Saltelli & Annoni, 2010). A new generation of FD-based methods is referred to as derivative-based global sensitivity measures (DGSM) (Kucherenko & Song, 2016; Sobol & Kucherenko, 2010; Kucherenko & Iooss, 2016) which average derivatives at points obtained via random or quasi-random sampling. Variogram analysis of response surfaces (Razavi & Gupta, 2016) is a framework to compute sensitivity indices based on the variance of FDs with equal distance across the input space. Gupta & Razavi (2018) present a global sensitivity matrix consisting of derivatives w.r.t. each input for multiple time steps in dynamic systems.

Distribution-based methods aim to capture changes in the output distribution, often focusing on statistical moments such as the output variance (referred to as variance-based SA). In order to attribute the output variance to input effects of increasing order, the function to be evaluated is first additively decomposed into a high-dimensional model representation (HDMR) (Hoeffding, 1948; Sobol, 1990; Saltelli et al., 2008). The fraction of explained variance by individual terms within the HDMR is referred to as the Sobol index (Sobol, 1990). A link between DGSM and Sobol indices is demonstrated by Kucherenko & Song (2016). Variance-based sensitivity indices can be estimated in various ways (Puy et al., 2021). Recent efforts have focused on using Shapley values for variance-based SA, which can also be used for dependent inputs (Owen, 2014). Contribution to the sample mean (Bolado-Lavin et al., 2009) and variance (Tarantola et al., 2012) plots visualize quantile-wise effects of inputs on the model output mean and variance. A common critique is that the simplification of the output distribution to a single metric such as the mean or variance entails an unjustifiable loss in information. Moment-independent techniques (which are also referred to as distribution-based methods by some authors) aim to capture changes in the entire output distribution and relate them to changes in input variables (Chun et al., 2000; Borgonovo et al., 2012).

Regression-based methods are restricted to data-driven modeling. They either utilize some model-specific attribute, e.g., model coefficients, or evaluate the model fit w.r.t. changes in inputs, e.g., by excluding variables. For linear regression models, standardized correlation coefficients and partial correlation coefficients (which control for confounding variables) provide a natural sensitivity metric (Sudret, 2008).

Emulators: SA puts major focus on emulators or metamodels which approximate the model but are cheaper to evaluate. Such approximations are advantageous if the model is costly to evaluate and many model evaluations are needed such as

in variance-based SA. Popular methods include Gaussian processes (Le Gratiet et al., 2017; Marrel et al., 2008; 2009) and polynomial chaos expansion (Le Gratiet et al., 2017; Sudret, 2008). Introducing a metamodel requires accounting for additional uncertainty in the metamodel itself and its estimation (Razavi et al., 2021).

4. Bridging the Gap Between ML and SA

Evidently, there is a certain overlap between ML and SA: both fields are concerned with the explanation of input-output relationships, and many methods are used and developed concurrently, e.g., the FANOVA, model-specific explanations, evaluating changes in the model fit with varying features, or emulators. **Our position is that IML can be seen as a form of SA applied to ML.** Whereas SA is an explanation framework to analyze input-output relationships in virtually any complex system, we now analyze input-output relationships in a generalized ML system. We now formally describe this ML system and discuss how existing methods relate to this perspective and to each other.

4.1. Viewing IML as a Form of SA Applied to ML

Recall that in the SA (or systems modeling) sense, a model represents a function within the system. For the ML system we are modeling, this applies to the functions τ, c, $\mathcal{I}$, and Γ which we defined in Section 3. For instance, for c, we have that $\phi = c, \boldsymbol{z} = \boldsymbol{\lambda}$, and $\boldsymbol{q} = \widehat{\text{GE}}$; for Γ, we have that $\phi = \Gamma, \boldsymbol{z} = (\widehat{f}, \boldsymbol{\eta}, \mathcal{D}_{\text{explain}})$, and $\boldsymbol{q} = \Xi$.

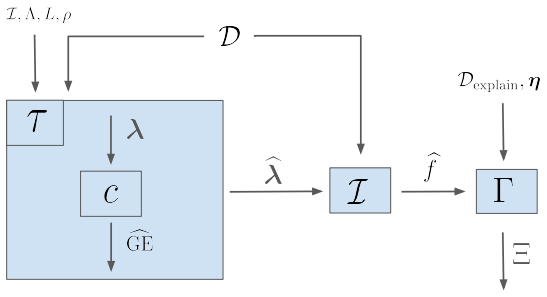


Figure 2. Formalizing the ML process as a system suitable for SA. The system consists of interconnected functions (indicated by boxes) of inputs and outputs.

Fig. 2 visualizes the ML system: functions correspond to boxes that receive inputs and produce outputs. This representation now enables us to view any explanation of the system under the common framework of SA. We can see that there is a cascading or "trickle-down" effect: choosing different system inputs trickles down through multiple functions and results in various outputs throughout the system. This might appear as an obvious fact, e.g., that choosing different hyperparameters results in a different model and consequently, in different model explanations, although the model is evaluated with the same data. **However, to formally describe sensitivities between different variables in ML, we first need to explicitly model their relationships.** In a certain sense, this is an effort to formalize a general theory of interpretations in ML.

IML Methods: Recall that IML methods are formulated to operate on two levels: the model and the hyperparameter level. In the ML system, this corresponds to:

- Γ: ICE, PD, FANOVA, ALE, PFI, counterfactual explanations, LIME, Shapley values
- c: FANOVA (Hutter et al., 2014), PD (Moosbauer et al., 2021)

Methods such as the PD and FANOVA have been used to explain the model and HPO process. After having presented a theory for SA within the ML system, this should not come as a surprise: every IML method simply computes sensitivities for input-output relationships (effects of features on the prediction, effects of features on the performance, or effects of hyperparameters on the performance) and could potentially be applied to other functions within the system. Furthermore, we can simply incorporate these novel IML methods into the SA toolbox and use them to compute sensitivities for entirely different systems, which, for instance, consist of physics-based models.

SA Methods: One could potentially apply numerous other SA techniques to ML, which are typically used to explain models in fields associated with SA. This includes but is not limited to the Morris method, DGSM, the global sensitivity matrix, variograms / variogram analysis of response surfaces, or Sobol index estimators (Kuhnt & Kalka, 2022; Stein et al., 2022; Paleari et al., 2021; Tunkiel et al., 2020; Ojha et al., 2022; Fel et al., 2021).

Additional Input Parameters for SA: In addition to using new methods to interpret the ML system, new interpretations can be created by considering different input parameters. Such suggestions are already often made in the context of specific methods, e.g., Bansal et al. (2020) analyze the sensitivity of model explanations for image classifiers regarding input parameters to the explanation method. Such novel types of SA for ML are now explicitly formalized in the context of this paper (with the explanation hyperparameter vector $\boldsymbol{\eta}$ dictating how Γ operates) and can easily be put into practice. For instance, we could evaluate the sensitivity of the PD curve when selecting different subsets of observations within $\mathcal{D}_{\text{explain}}$ or the sensitivity of the surrogate prediction for a single instance $\boldsymbol{x}$ for variations in the width parameter for LIME.

Dependent Features and Adjustments: Traditional SA is often based on process-based models where the system can (in good faith) be queried with input space-filling designs. In ML, the model is typically trained with dependent features, resulting in many areas of the feature space where the model has not seen much or any data. Model predictions in such areas (typically referred to as extrapolations) do not reflect any underlying data-generating process. Furthermore, in variance-based SA, there is an additional error source for dependent features: the HDMR can be non-unique, and there are non-zero covariance terms in the variance decomposition (Chastaing et al., 2013).

One might argue that in a model diagnostic sense, extrapolations do not pose a problem, as they represent accurate model predictions. Although there is some justification for this viewpoint, one might ask what the value behind a model and its explanations is if the model does not reflect any underlying real-world phenomenon. There are, however, potential solutions to this problem, at least for some methods: EEs or derivatives can, for instance, only be computed in high-density regions of the training set.

4.2. Explanations on the Data Level

Instead of evaluating function behavior, both SA and ML provide techniques that operate on the data directly. There are a few techniques that relate to traditional SA, e.g., SA on given-data (Plischke et al., 2013) (which aims to estimate sensitivity indices without querying a model) or traditional ML, e.g., the PD through stratification (Parr & Wilson, 2021) (which aims to do the same for the PD). Also referred to as green SA (Razavi et al., 2021), such sample-free approaches significantly reduce computational costs. Most data level techniques simply refer to some form of exploratory data analysis. For instance, scatter plots are used by the SA (Saltelli et al., 2008) and ML (Hastie et al., 2001) communities for simple analyses.

4.3. Model-Specific SA and Artificial Neural Networks

Model-specific interpretations typically represent some notion of sensitivity: for instance, tree splits tell us about changes in the average target value (the sensitivity of the target) when partitioning the data into subsets (w.r.t. inputs) for CART (Breiman et al., 1984); a beta coefficient for regression models informs us about the exact change in predicted outcome when adjusting feature values in a certain way. The nature of artificial neural networks (ANNs) allows for the design of powerful model-specific explanation techniques. Gradients can be more efficiently computed using symbolic derivatives which has resulted in numerous explanation methods (Ancona et al., 2017; Pizarroso et al., 2022). ANNs have proven especially well-suited for unstructured data such as image, text, or speech data. Many ANN-specific explanation techniques have been designed for such data as well, e.g., saliency maps (Simonyan et al., 2014) visualize the sensitivity of the predicted target w.r.t. the color values of a pixel. The term SA is often related to the analysis of ANNs (Yeung et al., 2010; Zhang & Wallace, 2016; Pizarroso et al., 2022; Mrzygłód et al., 2020), much more so than in the context of analyzing other ML models. Ablation studies for ANNs (Meyes et al., 2019) analyze how the removal of certain components affects the model performance. As the range of model types and corresponding model-specific explanations is vast, they are not further discussed in this paper. However, we stress the importance of model-specific SA, if applicable.

4.4. Further Contributions of SA

Apart from providing methodological advances, SA can contribute to ML in a variety of other ways.

Best Practices: The SA community has given considerable thought to what constitutes a high-quality analysis, which is typically termed sensitivity auditing (Saltelli et al., 2013; Lo Piano et al., 2022). This includes answering questions on what the evaluated function or model is used for, what the assumptions are, reproducibility, or the viewpoint of stakeholders. Although such questions are also discussed in ML contexts, they have not been compiled and formalized in the same way as in SA. Furthermore, SA defines formal settings and definitions such as factor fixing and factor prioritization which are tied to setting-suitable metrics such as certain variants of Sobol indices. In ML, interpretation concepts are poorly defined and are often determined by the interpretation method itself. For instance, there is no definition of a ground-truth feature importance, and importance scores produced by different methods often cannot be compared.

Application Workflows: In contrast to IML, the body of research in SA extensively discusses application workflows. For instance, it is common practice to screen for important input factors with computationally cheap methods first, e.g., using the Morris method or DGSM, before resorting to more accurate but computationally demanding techniques such as variance-based SA (Saltelli et al., 2008). As opposed to the research-focused, isolated development and evaluation of interpretation methods in ML, SA is much more embedded into industry applications and large-scale software systems. This has led to certain research directions that have been ignored in IML: for instance, Sheikholeslami et al. (2019a) explored strategies of grouping inputs to reduce computational costs; Sheikholeslami et al. (2019b) developed strategies to handle simulation crashes, e.g., due to numerical instabilities, without having to rerun the entire computer experiment.

4.5. Contributions of ML to SA

ML also significantly contributes to the advancement of SA. Notable contributions include better metamodeling practices through novel ML models, dependence measures and kernel-based indices, and Shapley values (Razavi et al., 2021).

In some fields such as hydrology, there still is a preference for theory-driven process-based modeling, which is often outperformed by modern ML models (Nearing et al., 2021). Apart from providing better predictive performance in data-driven modeling, this indicates that the current understanding of certain real-world phenomena is insufficient and that there is a potential to ultimately create better theory-driven models. Major efforts are put into merging process-based and data-driven modeling (sometimes referred to as hybrid modeling) (Razavi, 2021; Reichstein et al., 2019). For instance, in one-way coupling, the output of a mechanistic model represents the input of an ML model; in modular coupling, system models are either created from process-based or data-driven modeling based on which approach better suits the sub-modeling task (Razavi, 2021).

Furthermore, common constraints encountered in ML such as feature correlations have necessitated the development of interpretation methods able to handle these constraints, e.g., Shapley values. As noted by Razavi et al. (2021), such developments are still immature in SA, which can profit from novelties in ML. There is a point to be made that every novel interpretation method designed in an ML context is also applicable to any other mathematical model (or system), e.g., physics-based ones. This is restricted to IML methods that only need access to predictions, which is the case for the majority of methods. For instance, we imagine that ICEs, the PD, ALE, LIME, or Shapley values can provide tremendous value in interpreting various non-ML systems.

5. Discussion

In the following, major points for discussion shall be debated in a neutral light:

Why Should the ML Community Care About a Field With a Small Visibility? This might be the reason that the ML community has given little attention to related work in SA. We would like to bring forward three arguments here: First, a sound scientific process should strive towards proper crediting related work and avoiding redundancies; second, viewing IML as a form of SA applied to ML clarifies how we think about existing methods and lets us establish a common framework and terminology to discuss and formulate methods (which, besides, can be argued irrespectively of SA as an independent discipline); third, recent developments demonstrate an untapped potential of applying existing SA methods to ML (Kuhnt & Kalka, 2022; Stein et al., 2022; Paleari et al., 2021; Tunkiel et al., 2020; Ojha et al., 2022).

Is Every Method Included in This Framework? We formulated a general system of ML suitable for SA that includes common model-agnostic methods (Scholbeck et al., 2020) and novel methods to explain the HPO process, which have been directly derived from the SA literature, namely the FANOVA and the PD. Furthermore, we argued that this perspective includes model-specific interpretations such as tree splits, regression parameters, or ANN-specific methods. Even though this framework holds with great generality, we do not claim that every conceivable interpretation of the ML process is included. For the reasons we discussed in the previous paragraph, this does not, however, diminish the value of a unified SA-based perspective on IML.

What About Causality? The question of whether a feature causes a change in the actual outcome and not just in the predicted outcome carries increasing significance for ML and cannot be answered based on the predictive model alone; additional assumptions are needed, e.g., in the form of a causal graph (Molnar et al., 2022). Even if the model can perfectly predict the actual outcome, the causal effect can still run in both directions. Considerable effort has been put into the SA of causal inference (CI) models, e.g., to assess how robust associations are regarding unmeasured or uncontrolled confounding (VanderWeele & Ding, 2017; Veitch & Zaveri, 2020; Frauen et al., 2023). Due to the complexity of CI and the limited scope of application (causality is only relevant in some applications whereas predictions are always relevant in ML), CI is not further discussed here. However, the relevance of SA for CI in the literature demonstrates that there is a potential overlap to be explored in the future.

What About Unsupervised Learning? We factored out unsupervised learning (UL), including clustering methods, from our analysis. This stems from the fact that UL has been mostly ignored by the recent trend in interpretability research. Notably, a few novel works explore SA-inspired methods for algorithm-agnostic cluster explanations, e.g., L2PC and G2PC (Ellis et al., 2021) and feature attributions for clustering (FACT) (Scholbeck et al., 2023). This further underpins our argument that SA is a unifying framework for interpretations in ML and may include UL. Future research in UL explanations will reveal their relationship with SA.

6. Conclusion

This paper aims to direct attention to the concurrent development of similar approaches to interpret mathematical models in multiple communities. Our position is that IML can be seen as a form of SA applied to ML, which integrates recent advances in IML into a larger body of research on how to explain complex systems. To further substantiate our claim, we formalize the ML process as a system suitable for SA and discuss how existing methods relate to this perspective. With this paper, we strive towards a better recognition of

related work in the research community and the exploitation of potential research gaps.

Some readers might agree, and some might disagree with our viewpoint. The nature of a position paper is to initiate a discussion, and if we achieve a change of direction within the ML community as outlined above, this paper will have fulfilled its purpose.

References

Ancona, M., Ceolini, E., Öztireli, C., and Gross, M. Towards better understanding of gradient-based attribution methods for Deep Neural Networks. arXiv, 2017. URL https://doi.org/10.48550/arXiv.1711.06104.

Apley, D. W. and 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. URL https://doi.org/10.1111/rssb.12377.

Au, Q., Herbinger, J., Stachl, C., Bischl, B., and Casalicchio, G. Grouped feature importance and combined features effect plot. *Data Mining and Knowledge Discovery*, 36 (4):1401–1450, 2022. URL https://doi.org/10.1007/s10618-022-00840-5.

Ballester-Ripoll, R., Paredes, E. G., and Pajarola, R. Sobol tensor trains for global sensitivity analysis. *Reliability Engineering & System Safety*, 183:311–322, 2019. URL https://doi.org/10.1016/j.ress.2018.11.007.

Bansal, N., Agarwal, C., and Nguyen, A. SAM: The sensitivity of attribution methods to hyperparameters. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8670–8680. IEEE Computer Society, Los Alamitos, CA, USA, 2020. URL https://doi.org/10.1109/CVPR42600.2020.00870.

Baur, D., Cariboni, J., and Campolongo, F. Global sensitivity analysis for latent factor credit risk models. *International Journal of Risk Assessment and Management*, 11, 2004. URL https://doi.org/10.2139/ssrn.638563.

Becker, W., Rowson, J., Oakley, J., Yoxall, A., Manson, G., and Worden, K. Bayesian sensitivity analysis of a model of the aortic valve. *Journal of Biomechanics*, 44 (8):1499–1506, 2011. URL https://doi.org/10.1016/j.jbiomech.2011.03.008.

Bhangale, K. B. and Mohanaprasad, K. A review on speech processing using machine learning paradigm. *International Journal of Speech Technology*, 24(2):367–388, 2021. URL https://doi.org/10.1007/s10772-021-09808-0.

Biedenkapp, A., Lindauer, M., Eggensperger, K., Hutter, F., Fawcett, C., and Hoos, H. Efficient parameter importance analysis via ablation with surrogates. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1), 2017. URL https://doi.org/10.1609/aaai.v31i1.10657.

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A.-L., Deng, D., and Lindauer, M. Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *WIREs Data Mining and Knowledge Discovery*, 13(2):e1484, 2023. URL https://doi.org/10.1002/widm.1484.

Bolado-Lavin, R., Castaings, W., and Tarantola, S. Contribution to the sample mean plot for graphical and numerical sensitivity analysis. *Reliability Engineering & System Safety*, 94(6):1041–1049, 2009. URL https://doi.org/10.1016/j.ress.2008.11.012.

Borgonovo, E. and Plischke, E. Sensitivity analysis: A review of recent advances. *European Journal of Operational Research*, 248(3):869–887, 2016. URL https://doi.org/10.1016/j.ejor.2015.06.032.

Borgonovo, E., Castaings, W., and Tarantola, S. Model emulation and moment-independent sensitivity analysis: An application to environmental modelling. *Environmental Modelling and Software*, 34:105–115, 2012. URL https://doi.org/10.1016/j.envsoft.2011.06.006.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

Campolongo, F., Cariboni, J., and Saltelli, A. An effective screening design for sensitivity analysis of large models. *Environmental Modelling and Software*, 22:1509–1518, 2007.

Casalicchio, G., Molnar, C., and Bischl, B. Visualizing the feature importance for black box models. In Berlingerio, M., Bonchi, F., Gärtner, T., Hurley, N., and Ifrim, G. (eds.), *Machine Learning and Knowledge Discovery in Databases*, volume 11051 of *Lecture Notes in Computer Science*, pp. 655–670, Cham, 2019. Springer International Publishing. URL https://doi.org/10.1007/978-3-030-10925-7_40.

Chastaing, G., Prieur, C., and Gamboa, F. Generalized sobol sensitivity indices for dependent variables: Numerical methods. *Journal of Statistical Computation and*

Simulation, 85, 2013. URL https://doi.org/10.1080/00949655.2014.960415.

Chun, M.-H., Han, S.-J., and Tak, N.-I. An uncertainty importance measure using a distance metric for the change in a cumulative distribution function. *Reliability Engineering & System Safety*, 70(3):313–321, 2000.

Dandl, S., Molnar, C., Binder, M., and Bischl, B. Multi-objective counterfactual explanations. In Bäck, T., Preuss, M., Deutz, A., Wang, H., Doerr, C., Emmerich, M., and Trautmann, H. (eds.), *Parallel Problem Solving from Nature – PPSN XVI*, volume 12269 of *Lecture Notes in Computer Science*, pp. 448–469, Cham, 2020. Springer International Publishing.

Ellis, C. A., Sendi, M. S. E., Geenjaar, E. P. T., Plis, S. M., Miller, R. L., and Calhoun, V. D. Algorithm-agnostic explainability for unsupervised clustering. arXiv, 2021. URL https://doi.org/10.48550/arXiv.2105.08053.

Fawcett, C. and Hoos, H. H. Analysing differences between algorithm configurations through ablation. *Journal of Heuristics*, 22(4):431–458, 2016. URL https://doi.org/10.1007/s10732-014-9275-9.

Fel, T., Cadene, R., Chalvidal, M., Cord, M., Vigouroux, D., and Serre, T. Look at the variance! Efficient black-box explanations with Sobol-based sensitivity analysis. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=hA-PHQGOjqQ.

Feurer, M. and Hutter, F. Hyperparameter optimization. In Hutter, F., Kotthoff, L., and Vanschoren, J. (eds.), *Automated Machine Learning: Methods, Systems, Challenges*, pp. 3–33. Springer International Publishing, Cham, 2019. URL https://doi.org/10.1007/978-3-030-05318-5_1.

Fisher, A., Rudin, C., and 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.

Frauen, D., Imrie, F., Curth, A., Melnychuk, V., Feuerriegel, S., and van der Schaar, M. A neural framework for generalized causal sensitivity analysis. arXiv, 2023. URL https://doi.org/10.48550/arXiv.2311.16026.

Friedman, J. H. Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5):1189–1232, 2001.

Friedman, J. H. and Popescu, B. E. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2 (3), 2008. URL http://dx.doi.org/10.1214/07-AOAS148.

Gao, Y., Sahin, A., and Vrugt, J. A. Probabilistic sensitivity analysis with dependent variables: Covariance-based decomposition of hydrologic models. *Water Resources Research*, 59(4):e2022WR032834, 2023. URL https://doi.org/10.1029/2022WR032834.

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. 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. URL https://doi.org/10.1080/10618600.2014.907095.

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F., and Pedreschi, D. A survey of methods for explaining black box models. *ACM Comput. Surv.*, 51(5), 2018. URL https://doi.org/10.1145/3236009.

Guo, L., Meng, Z., Sun, Y., and Wang, L. Parameter identification and sensitivity analysis of solar cell models with cat swarm optimization algorithm. *Energy Conversion and Management*, 108:520–528, 2016. URL https://doi.org/10.1016/j.enconman.2015.11.041.

Gupta, H. V. and Razavi, S. Revisiting the basis of sensitivity analysis for dynamical earth system models. *Water Resources Research*, 54(11):8692–8717, 2018. URL https://doi.org/10.1029/2018WR022668.

Haghnegahdar, A. and Razavi, S. Insights into sensitivity analysis of earth and environmental systems models: On the impact of parameter perturbation scale. *Environmental Modelling & Software*, 95:115–131, 2017. URL https://doi.org/10.1016/j.envsoft.2017.03.031.

Harenberg, D., Marelli, S., Sudret, B., and Winschel, V. Uncertainty quantification and global sensitivity analysis for economic models. *Quantitative Economics*, 10(1): 1–41, 2019. URL https://doi.org/10.3982/QE866.

Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

Heavens, N. G., Ward, D. S., and Natalie, M. Studying and projecting climate change with earth system models. *Nature Education Knowledge*, 4(5):4, 2013.

Hoeffding, W. A class of statistics with asymptotically normal distribution. *The Annals of Mathematical Statistics*, 19(3):293 – 325, 1948. URL https://doi.org/10.1214/aoms/1177730196.

Homma, T. and Saltelli, A. Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety*, 52(1):1–17, 1996. URL https://doi.org/10.1016/0951-8320(96)00002-6.

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 '04, pp. 575–580, New York, NY, USA, 2004. ACM.

Hooker, G. Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables. *Journal of Computational and Graphical Statistics*, 16 (3):709–732, 2007.

Hooker, G., Mentch, L., and 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 (6):82, 2021. URL https://doi.org/10.1007/s11222-021-10057-z.

Huang, J., Chai, J., and Cho, S. Deep learning in finance and banking: A literature review and classification. *Frontiers of Business Research in China*, 14 (1):13, 2020. URL https://doi.org/10.1186/s11782-020-00082-6.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. Sequential model-based optimization for general algorithm configuration. In Coello, C. A. C. (ed.), *Learning and Intelligent Optimization - 5th International Conference, LION 5, Rome, Italy, January 17-21, 2011. Selected Papers*, volume 6683 of *Lecture Notes in Computer Science*, pp. 507–523. Springer, 2011. URL https://doi.org/10.1007/978-3-642-25566-3_40.

Hutter, F., Hoos, H., and Leyton-Brown, K. An efficient approach for assessing hyperparameter importance. *31st International Conference on Machine Learning, ICML 2014*, 2:1130–1144, 2014.

Iooss, B. and Lemaître, P. A review on global sensitivity analysis methods. In *Uncertainty Management in Simulation-Optimization of Complex Systems: Algorithms and Applications*, pp. 101–122. Springer US, Boston, MA, 2015. URL https://doi.org/10.1007/978-1-4899-7547-8_5.

Jones, D. R., Schonlau, M., and Welch, W. J. Efficient global optimization of expensive black-box functions. *J. Glob. Optim.*, 13(4):455–492, 1998. URL https://doi.org/10.1023/A:1008306431147.

Kawamiya, M., Hajima, T., Tachiiri, K., Watanabe, S., and Yokohata, T. Two decades of earth system modeling with an emphasis on model for interdisciplinary research on climate (MIROC). *Progress in Earth and Planetary Science*, 7(1):64, 2020. URL https://doi.org/10.1186/s40645-020-00369-5.

Kucherenko, S. and Iooss, B. Derivative-based global sensitivity measures. In *Handbook of Uncertainty Quantification*, pp. 1–24. Springer International Publishing, Cham, 2016. URL https://doi.org/10.1007/978-3-319-11259-6_36-1.

Kucherenko, S. and Song, S. Derivative-based global sensitivity measures and their link with Sobol' sensitivity indices. *Monte Carlo and Quasi-Monte Carlo Methods*, pp. 455–469, 2016. URL http://dx.doi.org/10.1007/978-3-319-33507-0_23.

Kuhnt, S. and Kalka, A. Global sensitivity analysis for the interpretation of machine learning algorithms. In Steland, A. and Tsui, K.-L. (eds.), *Artificial Intelligence, Big Data and Data Science in Statistics: Challenges and Solutions in Environmetrics, the Natural Sciences and Technology*, pp. 155–169. Springer International Publishing, Cham, 2022. URL https://doi.org/10.1007/978-3-031-07155-3_6.

Le Gratiet, L., Marelli, S., and Sudret, B. Metamodel-based sensitivity analysis: Polynomial chaos expansions and Gaussian processes. In *Handbook of Uncertainty Quantification*, pp. 1289–1325. Springer International Publishing, Cham, 2017. URL https://doi.org/10.1007/978-3-319-12385-1_38.

Lo Piano, S., Sheikholeslami, R., Puy, A., and Saltelli, A. Unpacking the modelling process via sensitivity auditing. *Futures*, 144:103041, 2022. URL https://doi.org/10.1016/j.futures.2022.103041.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, pp. 4765–4774. Curran Associates, Inc., 2017.

Mai, J., Craig, J. R., Tolson, B. A., and Arsenault, R. The sensitivity of simulated streamflow to individual hydrologic processes across North America. *Nature Communications*, 13(1):455, 2022. URL https://doi.org/10.1038/s41467-022-28010-7.

Marrel, A., Iooss, B., Van Dorpe, F., and Volkova, E. An efficient methodology for modeling complex computer codes with Gaussian processes. *Computational Statistics & Data Analysis*, 52:4731–4744, 2008. URL https://doi.org/10.1016/j.csda.2008.03.026.

Marrel, A., Iooss, B., Laurent, B., and Roustant, O. Calculations of Sobol indices for the Gaussian process metamodel. *Reliability Engineering & System Safety*, 94:742–751, 2009. URL https://doi.org/10.1016/j.ress.2008.07.008.

Meyes, R., Lu, M., de Puiseau, C. W., and Meisen, T. Ablation studies in artificial neural networks. arXiv, 2019. URL https://doi.org/10.48550/arXiv.1901.08644.

Molnar, C. *Interpretable Machine Learning*. 2nd edition, 2022. URL https://christophm.github.io/interpretable-ml-book.

Molnar, C., Casalicchio, G., and Bischl, B. Interpretable machine learning – A brief history, state-of-the-art and challenges. In Koprinska, I., Kamp, M., Appice, A., Loglisci, C., Antonie, L., Zimmermann, A., Guidotti, R., Özgöbek, Ö., Ribeiro, R. P., Gavaldà, R., Gama, J., Adilova, L., Krishnamurthy, Y., Ferreira, P. M., Malerba, D., Medeiros, I., Ceci, M., Manco, G., Masciari, E., Ras, Z. W., Christen, P., Ntoutsi, E., Schubert, E., Zimek, A., Monreale, A., Biecek, P., Rinzivillo, S., Kille, B., Lommatzsch, A., and Gulla, J. A. (eds.), *ECML PKDD 2020 Workshops*, pp. 417–431, Cham, 2020. Springer International Publishing. URL https://doi.org/10.1007/978-3-030-65965-3_28.

Molnar, C., König, G., Herbinger, J., Freiesleben, T., Dandl, S., Scholbeck, C. A., Casalicchio, G., Grosse-Wentrup, M., and Bischl, B. 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. (eds.), *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*, pp. 39–68. Springer International Publishing, Cham, 2022. URL https://doi.org/10.1007/978-3-031-04083-2_4.

Molnar, C., König, G., Bischl, B., and Casalicchio, G. Model-agnostic feature importance and effects with dependent features: A conditional subgroup approach. *Data Mining and Knowledge Discovery*, 38(5):2903–2941, 2024. URL https://doi.org/10.1007/s10618-022-00901-9.

Moosbauer, J., Herbinger, J., Casalicchio, G., Lindauer, M., and Bischl, B. Explaining hyperparameter optimization via partial dependence plots. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=k8KDqVbIS2l.

Morris, M. D. Factorial sampling plans for preliminary computational experiments. *Technometrics*, pp. 161–174, 1991.

Mrzygłód, B., Hawryluk, M., Janik, M., and Olejarczyk-Wożeńska, I. Sensitivity analysis of the artificial neural networks in a system for durability prediction of forging tools to forgings made of C45 steel. *The International Journal of Advanced Manufacturing Technology*, 109 (5):1385–1395, 2020. URL https://doi.org/10.1007/s00170-020-05641-y.

Nearing, G. S., Kratzert, F., Sampson, A. K., Pelissier, C. S., Klotz, D., Frame, J. M., Prieto, C., and Gupta, H. V. What role does hydrological science play in the age of machine learning? *Water Resources Research*, 57(3): e2020WR028091, 2021. URL https://doi.org/10.1029/2020WR028091.

Nossent, J., Elsen, P., and Bauwens, W. Sobol' sensitivity analysis of a complex environmental model. *Environmental Modelling & Software*, 26(12):1515–1525, 2011. URL https://doi.org/10.1016/j.envsoft.2011.08.010.

Ojha, V., Timmis, J., and Nicosia, G. Assessing ranking and effectiveness of evolutionary algorithm hyperparameters using global sensitivity analysis methodologies. *Swarm and Evolutionary Computation*, 74: 101130, 2022. URL https://doi.org/10.1016/j.swevo.2022.101130.

Owen, A. Sobol' indices and Shapley value. *SIAM/ASA Journal on Uncertainty Quantification*, 2:245–251, 2014. URL https://doi.org/10.1137/130936233.

Paleari, L., Movedi, E., Zoli, M., Burato, A., Cecconi, I., Errahouly, J., Pecollo, E., Sorvillo, C., and Confalonieri, R. Sensitivity analysis using Morris: Just screening or an effective ranking method? *Ecological Modelling*, 455: 109648, 2021. URL https://doi.org/10.1016/j.ecolmodel.2021.109648.

Parr, T. and Wilson, J. D. Partial dependence through stratification. *Machine Learning with Applications*, 6: 100146, 2021. URL https://doi.org/10.1016/j.mlwa.2021.100146.

Pianosi, F., Beven, K., Freer, J., Hall, J. W., Rougier, J., Stephenson, D. B., and Wagener, T. Sensitivity analysis of environmental models: A systematic review with

practical workflow. *Environmental Modelling & Software*, 79:214–232, 2016. URL https://doi.org/10.1016/j.envsoft.2016.02.008.

Pierson, H. A. and Gashler, M. S. Deep learning in robotics: A review of recent research. *Advanced Robotics*, 31 (16):821–835, 2017. URL https://doi.org/10.1080/01691864.2017.1365009.

Pizarroso, J., Portela, J., and Muñoz, A. Neuralsens: Sensitivity analysis of neural networks. *Journal of Statistical Software*, 102(7):1–36, 2022. URL https://doi.org/10.18637/jss.v102.i07.

Plischke, E., Borgonovo, E., and Smith, C. L. Global sensitivity measures from given data. *European Journal of Operational Research*, 226(3):536–550, 2013. URL https://doi.org/10.1016/j.ejor.2012.11.047.

Puy, A., Becker, W., Lo Piano, S., and Saltelli, A. A comprehensive comparison of total-order estimators for global sensitivity analysis. *International Journal for Uncertainty Quantification*, 2021. URL http://dx.doi.org/10.1615/Int.J.UncertaintyQuantification.2021038133.

Rabitz, H. and Aliş, Ö. F. General foundations of high-dimensional model representations. *Journal of Mathematical Chemistry*, 25(2):197–233, 1999. URL https://doi.org/10.1023/A:1019188517934.

Rajkomar, A., Dean, J., and Kohane, I. Machine learning in medicine. *New England Journal of Medicine*, 380 (14):1347–1358, 2019. URL https://doi.org/10.1056/NEJMra1814259.

Ratto, M. Analysing DSGE models with global sensitivity analysis. *Computational Economics*, 31(2):115–139, 2008. URL https://doi.org/10.1007/s10614-007-9110-6.

Razavi, S. Deep learning, explained: Fundamentals, explainability, and bridgeability to process-based modelling. *Environmental Modelling & Software*, 144:105159, 2021. URL https://doi.org/10.1016/j.envsoft.2021.105159.

Razavi, S. and Gupta, H. V. What do we mean by sensitivity analysis? The need for comprehensive characterization of "global" sensitivity in earth and environmental systems models. *Water Resources Research*, 51(5):3070–3092, 2015. URL https://doi.org/10.1002/2014WR016527.

Razavi, S. and Gupta, H. V. A new framework for comprehensive, robust, and efficient global sensitivity analysis: 1. Theory. *Water Resources Research*, 52(1):423–439, 2016. URL https://doi.org/10.1002/2015WR017558.

Razavi, S., Jakeman, A., Saltelli, A., Prieur, C., Iooss, B., Borgonovo, E., Plischke, E., Lo Piano, S., Iwanaga, T., Becker, W., Tarantola, S., Guillaume, J. H., Jakeman, J., Gupta, H., Melillo, N., Rabitti, G., Chabridon, V., Duan, Q., Sun, X., Smith, S., Sheikholeslami, R., Hosseini, N., Asadzadeh, M., Puy, A., Kucherenko, S., and Maier, H. R. The future of sensitivity analysis: An essential discipline for systems modeling and policy support. *Environmental Modelling and Software*, 137: 104954, 2021. URL https://doi.org/10.1016/j.envsoft.2020.104954.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., and Prabhat. Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204, 2019. URL https://doi.org/10.1038/s41586-019-0912-1.

Ribeiro, M. T., Singh, S., and 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*, KDD '16, pp. 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. URL https://doi.org/10.1145/2939672.2939778.

Rolnick, D., Donti, P. L., Kaack, L. H., Kochanski, K., Lacoste, A., Sankaran, K., Ross, A. S., Milojevic-Dupont, N., Jaques, N., Waldman-Brown, A., Luccioni, A. S., Maharaj, T., Sherwin, E. D., Mukkavilli, S. K., Kording, K. P., Gomes, C. P., Ng, A. Y., Hassabis, D., Platt, J. C., Creutzig, F., Chayes, J., and Bengio, Y. Tackling climate change with machine learning. *ACM Comput. Surv.*, 55(2), 2022. URL https://doi.org/10.1145/3485128.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 16:1 – 85, 2022. URL https://doi.org/10.1214/21-SS133.

Saltelli, A. and Annoni, P. How to avoid a perfunctory sensitivity analysis. *Environmental Modelling & Software*, 25(12):1508 – 1517, 2010. URL https://doi.org/10.1016/j.envsoft.2010.04.012.

Saltelli, A. and Tarantola, S. On the relative importance of input factors in mathematical models. *Journal of the American Statistical Association*, 97(459):702–709, 2002. URL https://doi.org/10.1198/016214502388618447.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. *Global Sensitivity Analysis: The Primer*. Wiley, 2008.

Saltelli, A., Guimaraes Pereira, Â., Sluijs, J. P. V. d., and Funtowicz, S. What do I make of your Latinorum? Sensitivity auditing of mathematical modelling. *International Journal of Foresight and Innovation Policy*, 9 (2/3/4):213, 2013. URL http://dx.doi.org/10.1504/IJFIP.2013.058610.

Saltelli, A., Aleksankina, K., Becker, W., Fennell, P., Ferretti, F., Holst, N., Li, S., and Wu, Q. Why so many published sensitivity analyses are false: A systematic review of sensitivity analysis practices. *Environmental Modelling and Software*, 114:29–39, 2019. URL https://doi.org/10.1016/j.envsoft.2019.01.012.

Scholbeck, C. A., Molnar, C., Heumann, C., Bischl, B., and Casalicchio, G. Sampling, intervention, prediction, aggregation: A generalized framework for model-agnostic interpretations. In Cellier, P. and Driessens, K. (eds.), *Machine Learning and Knowledge Discovery in Databases*, volume 1167 of *Communications in Computer and Information Science*, pp. 205–216, Cham, 2020. Springer International Publishing. URL https://doi.org/10.1007/978-3-030-43823-4_18.

Scholbeck, C. A., Funk, H., and Casalicchio, G. Algorithm-agnostic feature attributions for clustering. In Longo, L. (ed.), *Explainable Artificial Intelligence*, volume 1901 of *Communications in Computer and Information Science*, pp. 217–240, Cham, 2023. Springer Nature Switzerland. URL https://doi.org/10.1007/978-3-031-44064-9_13.

Sheikholeslami, R., Razavi, S., Gupta, H. V., Becker, W., and Haghnegahdar, A. Global sensitivity analysis for high-dimensional problems: How to objectively group factors and measure robustness and convergence while reducing computational cost. *Environmental Modelling & Software*, 111:282–299, 2019a. URL https://doi.org/10.1016/j.envsoft.2018.09.002.

Sheikholeslami, R., Razavi, S., and Haghnegahdar, A. What should we do when a model crashes? Recommendations for global sensitivity analysis of earth and environmental systems models. *Geoscientific Model Development*, 12 (10):4275–4296, 2019b. URL https://doi.org/10.5194/gmd-12-4275-2019.

Shin, M.-J., Guillaume, J. H., Croke, B. F., and Jakeman, A. J. Addressing ten questions about conceptual rainfall–runoff models with global sensitivity analyses in R. *Journal of Hydrology*, 503:135–152, 2013. URL https://doi.org/10.1016/j.jhydrol.2013.08.047.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop at International Conference on Learning Representations*, 2014.

Snoek, J., Larochelle, H., and Adams, R. P. Practical Bayesian optimization of machine learning algorithms. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'12, pp. 2951–2959, Red Hook, NY, USA, 2012. Curran Associates Inc.

Sobol, I. On sensitivity estimation for nonlinear mathematical models. *Matem. Mod.*, 2:112–118, 1990.

Sobol, I. and Kucherenko, S. Derivative based global sensitivity measures. *Procedia - Social and Behavioral Sciences*, 2(6):7745 – 7746, 2010. URL https://doi.org/10.1016/j.sbspro.2010.05.208. Sixth International Conference on Sensitivity Analysis of Model Output.

Song, X., Zhang, J., Zhan, C., Xuan, Y., Ye, M., and Xu, C. Global sensitivity analysis in hydrological modeling: Review of concepts, methods, theoretical framework, and applications. *Journal of Hydrology*, 523:739–757, 2015. URL https://doi.org/10.1016/j.jhydrol.2015.02.013.

Stein, B. V., Raponi, E., Sadeghi, Z., Bouman, N., Van Ham, R. C. H. J., and Bäck, T. A comparison of global sensitivity analysis methods for explainable AI with an application in genomic prediction. *IEEE Access*, 10:103364–103381, 2022. URL https://doi.org/10.1109/ACCESS.2022.3210175.

Stigler, S. M. Gauss and the invention of least squares. *The Annals of Statistics*, 9(3):465–474, 1981.

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

Sudret, B. Global sensitivity analysis using polynomial chaos expansion. *Reliability Engineering & System Safety*, 93:964–979, 2008. URL https://doi.org/10.1016/j.ress.2007.04.002.

Sumner, T., Shephard, E., and Bogle, I. D. L. A methodology for global-sensitivity analysis of time-dependent outputs in systems biology modelling. *J R Soc Interface*, 9(74):2156–2166, 2012.

Tarantola, S., Kopustinskas, V., Bolado-Lavin, R., Kaliatka, A., Uščuras, E., and Vaišnoras, M. Sensitivity analysis using contribution to sample variance plot: Application to a water hammer model. *Reliability Engineering & System Safety*, 99:62–73, 2012. URL https://doi.org/10.1016/j.ress.2011.10.007.

Tian, W. A review of sensitivity analysis methods in building energy analysis. *Renewable and Sustainable Energy Reviews*, 20(C):411–419, 2013. URL https://doi.org/10.1016/j.rser.2012.12.014.

Tunkiel, A. T., Sui, D., and Wiktorski, T. Data-driven sensitivity analysis of complex machine learning models: A case study of directional drilling. *Journal of Petroleum Science and Engineering*, 195: 107630, 2020. URL https://doi.org/10.1016/j.petrol.2020.107630.

VanderWeele, T. J. and Ding, P. Sensitivity analysis in observational research: Introducing the e-value. *Annals of Internal Medicine*, 167(4):268–274, 2017. URL https://doi.org/10.7326/M16-2607.

Vanschoren, J. Meta-learning: A survey. arXiv, 2018. URL https://doi.org/10.48550/arXiv.1810.03548.

Veitch, V. and Zaveri, A. Sense and sensitivity analysis: Simple post-hoc analysis of bias due to unobserved confounding. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc.

Wachter, S., Mittelstadt, B., and Russell, C. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law and Technology*, 31(2):841–887, 2018.

Wagener, T. and Pianosi, F. What has global sensitivity analysis ever done for us? A systematic review to support scientific advancement and to inform policy-making in earth system modelling. *Earth-Science Reviews*, 194: 1–18, 2019. URL https://doi.org/10.1016/j.earscirev.2019.04.006.

Wei, P., Lu, Z., and Song, J. Variable importance analysis: A comprehensive review. *Reliability Engineering & System Safety*, 142:399–432, 2015. URL https://doi.org/10.1016/j.ress.2015.05.018.

Woźnica, K. and Biecek, P. Towards explainable meta-learning. In Kamp, M., Koprinska, I., Bibal, A., Bouadi, T., Frénay, B., Galárraga, L., Oramas, J., Adilova, L., Krishnamurthy, Y., Kang, B., Largeron, C., Lijffijt, J., Viard, T., Welke, P., Ruocco, M., Aune, E., Gallicchio, C., Schiele, G., Pernkopf, F., Blott, M., Fröning, H., Schindler, G., Guidotti, R., Monreale, A., Rinzivillo, S., Biecek, P., Ntoutsi, E., Pechenizkiy, M., Rosenhahn, B., Buckley, C., Cialfi, D., Lanillos, P., Ramstead, M., Verbelen, T., Ferreira, P. M., Andresini, G., Malerba, D., Medeiros, I., Fournier-Viger, P., Nawaz, M. S., Ventura, S., Sun, M., Zhou, M., Bitetta, V., Bordino, I., Ferretti, A., Gullo, F., Ponti, G., Severini, L., Ribeiro, R., Gama, J., Gavaldà, R., Cooper, L., Ghazaleh, N., Richiardi, J., Roqueiro, D., Saldana Miranda, D., Sechidis, K., and Graça, G. (eds.), *Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, volume 1524 of *Communications in Computer and Information Science*, pp. 505–520, Cham, 2021. Springer International Publishing. URL https://doi.org/10.1007/978-3-030-93736-2_38.

Yeung, D., Cloete, I., Shi, D., and Ng, W. *Sensitivity Analysis for Neural Networks*. Springer-Verlag Berlin Heidelberg, 2010. URL https://doi.org/10.1007/978-3-642-02532-7.

Zhang, Y. and Wallace, B. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. arXiv, 2016. URL https://doi.org/10.48550/arXiv.1510.03820.

Zhang, Y., Chen, M., and Liu, L. A review on text mining. In *2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, pp. 681–685, 2015. URL https://doi.org/10.1109/ICSESS.2015.7339149.

8 | Marginal Effects for Non-Linear Prediction Functions

Contributing Paper

Scholbeck, C. A., Casalicchio, G., Molnar, C., Bischl, B., and Heumann, C. (2024). "Marginal Effects for Non-Linear Prediction Functions". In: *Data Mining and Knowledge Discovery* 38.5, pp. 2997–3042. DOI: 10.1007/s10618-023-00993-x

Declaration of Contributions

C.A. Scholbeck contributed to this paper as the first author. The project was conceptualized by C.A. Scholbeck and G. Casalicchio in close collaboration. C.A. Scholbeck developed the idea of using FMEs as a model-agnostic interpretation method, introduced the NLM, and developed the cAME with valuable input from, notably, G. Casalicchio and from C. Molnar, B. Bischl, and C. Heumann. C.A. Scholbeck wrote the software implementation; created all simulations, applied examples, and visualizations; developed all proofs; drafted the paper; and revised it according to the feedback from his co-authors and external reviewers.

G. Casalicchio initiated the project based on his prior research activities concerning DMEs. The idea of using a decision tree to find subgroups associated with homogeneous model explanations was initially developed by G. Casalicchio and B. Bischl with equal contributions. C. Heumann suggested to formulate averages of FMEs and NLM values as estimators of the underlying expected effects. G. Casalicchio, B. Bischl, and C. Heumann provided valuable support throughout multiple revisions and suggested several notable modifications.

Data Mining and Knowledge Discovery (2024) 38:2997–3042
https://doi.org/10.1007/s10618-023-00993-x

Marginal effects for non-linear prediction functions

Christian A. Scholbeck[1] · Giuseppe Casalicchio[1] · Christoph Molnar[2] · Bernd Bischl[1] · Christian Heumann[2]

Received: 31 March 2021 / Accepted: 23 November 2023 / Published online: 27 February 2024

Abstract

Beta coefficients for linear regression models represent the ideal form of an interpretable feature effect. However, for non-linear models such as generalized linear models, the estimated coefficients cannot be interpreted as a direct feature effect on the predicted outcome. Hence, marginal effects are typically used as approximations for feature effects, either as derivatives of the prediction function or forward differences in prediction due to changes in feature values. While marginal effects are commonly used in many scientific fields, they have not yet been adopted as a general model-agnostic interpretation method for machine learning models. This may stem from the ambiguity surrounding marginal effects and their inability to deal with the non-linearities found in black box models. We introduce a unified definition of forward marginal effects (FMEs) that includes univariate and multivariate, as well as continuous, categorical, and mixed-type features. To account for the non-linearity of prediction functions, we introduce a non-linearity measure for FMEs. Furthermore, we argue against summarizing feature effects of a non-linear prediction function in a single metric such as the average marginal effect. Instead, we propose to average homogeneous FMEs within population subgroups, which serve as conditional feature effect estimates.

Keywords Forward difference · Forward marginal effects · FME · NLM · cAME · Non-linearity measure · Conditional average marginal effects · Interpretable machine learning · Explainable AI · IML · XAI · Interpretations · Model-agnostic

Responsible editor: Martin Atzmüller, Johannes Fürnkranz, Tomas Kliegr, and Ute Schmid.

✉ Christian A. Scholbeck
christian.scholbeck@stat.uni-muenchen.de

Giuseppe Casalicchio
giuseppe.casalicchio@stat.uni-muenchen.de

[1] Munich Center for Machine Learning (MCML), Ludwig-Maximilians-Universität in Munich, Munich, Germany

[2] Ludwig-Maximilians-Universität in Munich, Munich, Germany

1 Introduction

The lack of interpretability of most machine learning (ML) models has been considered one of their major drawbacks (Breiman 2001b). As a consequence, researchers have developed a variety of model-agnostic techniques to explain the behavior of ML models. These techniques are commonly referred to by the umbrella terms of interpretable machine learning (IML) or explainable artificial intelligence. Model explanations take different forms, e.g., feature attributions (FAs) such as a value indicating a feature's importance to the model or a curve indicating its effects on the prediction, model internals such as beta coefficients for linear regression models, data points such as counterfactual explanations (Wachter et al. 2018), or surrogate models (i.e., interpretable approximations to the original model) (Molnar 2022). In the context of our paper, we categorize an FA as an effect or importance:

- **Feature effect:** We define a feature effect as the direction and magnitude of a change in predicted outcome due to a change in feature values (Casalicchio et al. 2019; Scholbeck et al. 2020).
- **Feature importance:** Importance is an indication of a feature's relevance to the model. Effect and importance are related, as a feature with a large effect on the prediction can also be considered important. However, a feature's relevance can be measured in multiple ways; for instance, the permutation feature importance (Fisher et al. 2019) shuffles feature values and evaluates changes in model performance, while the functional analysis of variance (Saltelli et al. 2008; Hooker 2004b, 2007) evaluates contributions of terms within a high-dimensional model representation to the model output variance.

In this paper, we focus on feature effects, which are relevant for many applications. We distinguish between local explanations on the observational level and global ones for the entire feature space. For example, in medical research, we might want to assess the increase in risk of contracting a disease due to a change in a patient's health characteristics such as age or body weight. Consider the interpretation of a linear regression model (LM) without interaction terms where β_j denotes the coefficient of the j-th feature. Increasing a feature value x_j by one unit causes a change in predicted outcome of β_j. LMs are therefore often interpreted by merely inspecting the estimated coefficients. When the terms are non-linear, interactions are present, or when the expected target is transformed such as in generalized linear models (GLMs), interpretations are both inconvenient and unintuitive. For instance, in logistic regression, the expectation of the target variable is logit-transformed, and the predictor term cannot be interpreted as a direct feature effect on the predicted risk. It follows that even linear terms have a non-linear effect on the predicted target that varies across the feature space and makes interpretations through the model parameters difficult to impossible. A more convenient and intuitive interpretation corresponds to the derivative of the prediction function w.r.t. the feature or inspecting the change in prediction

due to an intervention in the data. These two approaches are commonly referred to as marginal effects (MEs) in statistical literature (Bartus 2005). MEs are often aggregated to an average marginal effect (AME), which represents an estimate of the expected ME. Furthermore, marginal effects at means (MEM) and marginal effects at representative values (MER) correspond to MEs where all features are set to the sample mean or where some feature values are set to manually chosen values (Williams 2012). These can be used to answer common research questions, e.g., what the average effect of age or body weight is on the risk of contracting the disease (AME), what the effect is for a patient with average age and body weight (MEM), and what the effect is for a patient with pre-specified age and body weight values (MER). An increasing amount of scientific disciplines now rely on the predictive power of black box ML models instead of using intrinsically interpretable models such as GLMs, e.g., econometrics (Athey 2017) or psychology (Stachl et al. 2017). This creates an incentive to review and refine the theory of MEs for the application to non-linear models.

For one, there is much confusion regarding the definition of MEs, evidenced by two variants for continuous features (based on either derivatives or forward differences) and furthermore by categorical MEs (which are computed as finite differences resulting from switching categories in various ways). In their current form, MEs are not an ideal tool to interpret many statistical models such as GLMs, and their shortcomings are exacerbated when applied to black box models such as the ones created by many ML algorithms. For non-linear prediction functions, MEs based on derivatives provide misleading feature effect interpretations: Given the tangent to the prediction function at a point x, we evaluate the tangent's rise at a point $x + h$. A unit increase for h is typically used as an interpretable standard measure. For non-linear prediction functions however, this change in feature values results in a different prediction than implied by the derivative ME, thereby rendering this interpretation misleading. The alternative and often overlooked definition based on forward differences is much better suited for effect interpretations but also suffers from a loss in information about the shape of the prediction function (see Sect. 3). For linear models, the ME is identical across the entire feature space. For non-linear models, one typically estimates the global feature effect by computing the AME (Bartus 2005; Onukwugha et al. 2015). However, a global average does not accurately represent the nuances of a non-linear predictive model. A more informative summary of the prediction function corresponds to the conditional feature effect on a feature subspace, e.g., patients with an entire range of health characteristics might be associated with homogeneous feature effects. Instead of global interpretations on the entire feature space, one should instead aim for semi-aggregated (regional or semi-global) interpretations. More specifically, one should work towards computing multiple, regional conditional AMEs (cAMEs) instead of a single, global AME.

Contributions: This paper introduces forward marginal effects (FMEs) as a model-agnostic interpretation method for arbitrary prediction functions[1]. We first provide a unified definition of FMEs for both univariate and multivariate, as well as continuous, categorical, and mixed-type features. Then, we define a non-linearity measure (NLM) for FMEs based on the similarity between the prediction function and the intersecting linear secant. Furthermore, for a more nuanced interpretation, we introduce conditional AMEs (cAMEs) for population subgroups as a regional (semi-global) feature effect measure that more accurately describes feature effects across the feature space. We propose one option to find subgroups for cAMEs by recursively partitioning the feature space with a regression tree on FMEs. Furthermore, we provide proofs on additive recovery for the univariate and multivariate FME and a proof on the relation between the individual conditional expectation (ICE) / partial dependence (PD) and the FME / forward AME.

Structure of the paper: In Sect. 2, we introduce our notation. In Sect. 3, we make sense of the ambiguous usage of MEs. In Sect. 4, we introduce a unified definition of FMEs, the NLM, and the cAME. Section 5 provides an overview on related work, demonstrates the relation between FMEs and the ICE / PD, and compares FMEs to the competing approach LIME. In Sect. 6, we run multiple simulations showcasing FMEs and the NLM. In Sect. 7, we present a structured application workflow and an applied example on real data. The Appendix contains background information on additive decompositions of prediction functions, on model extrapolations, on MEs for tree-based functions, as well as the above-mentioned mathematical proofs.

2 Notation

We consider a p-dimensional feature space $\mathscr{X} = \mathscr{X}_1 \times \cdots \times \mathscr{X}_p$ and a target space $\mathscr{Y}$. The random variables on the feature space are denoted by $\boldsymbol{X} = (X_1, \ldots, X_p)$.[2] The random variable on the target space is denoted by Y. A generic subspace of all features is denoted by $\mathscr{X}_{[\,]} \subseteq \mathscr{X}$. Correspondingly, $\boldsymbol{X}$ with a restricted sample space is denoted by $\boldsymbol{X}_{[\,]}$. A realization of $\boldsymbol{X}$ and Y is denoted by $\boldsymbol{x} = (x_1, \ldots, x_p)$ and y. The probability distribution $\mathscr{P}$ is defined on the sample space $\mathscr{X} \times \mathscr{Y}$. A learning algorithm trains a predictive model $\widehat{f} : \mathbb{R}^p \mapsto \mathbb{R}$ on data drawn from $\mathscr{P}$, where $\widehat{f}(\boldsymbol{x})$ denotes the model prediction based on the p-dimensional feature vector $\boldsymbol{x}$. To simplify our notation, we only consider one-dimensional predictions. However, the results on MEs can be generalized to multi-dimensional predictions, e.g., for multi-class classification. We denote the value of the j-th feature in $\boldsymbol{x}$ by x_j. A set of features is denoted by $S \subseteq \{1, \ldots, p\}$. The values of the feature set are denoted by $\boldsymbol{x}_S$.[3] All complementary features are indexed by $-j$ or $-S$, so that $\boldsymbol{x}_{-j} = \boldsymbol{x}_{\{1,\ldots,p\} \setminus \{j\}}$, or $\boldsymbol{x}_{-S} = \boldsymbol{x}_{\{1,\ldots,p\} \setminus S}$. An instance $\boldsymbol{x}$ can be partitioned so that $\boldsymbol{x} = (x_j, \boldsymbol{x}_{-j})$, or $\boldsymbol{x} = (\boldsymbol{x}_S, \boldsymbol{x}_{-S})$. With slight abuse of notation, we may denote the vector $\boldsymbol{x}_S$ by $(x_1, \ldots, x_s)$ regardless

[1] During the peer review process, we began to implement the theory presented in this manuscript in the R package `fmeffects` (Löwe et al. 2023)

[2] Vectors are denoted in bold letters.

[3] As $\boldsymbol{x}_S$ is the generalization of x_j to vectors, we denote it in bold letters. However, it can in fact be a scalar. The same holds for $\boldsymbol{x}_{-S}$ and $\boldsymbol{x}_{-j}$.

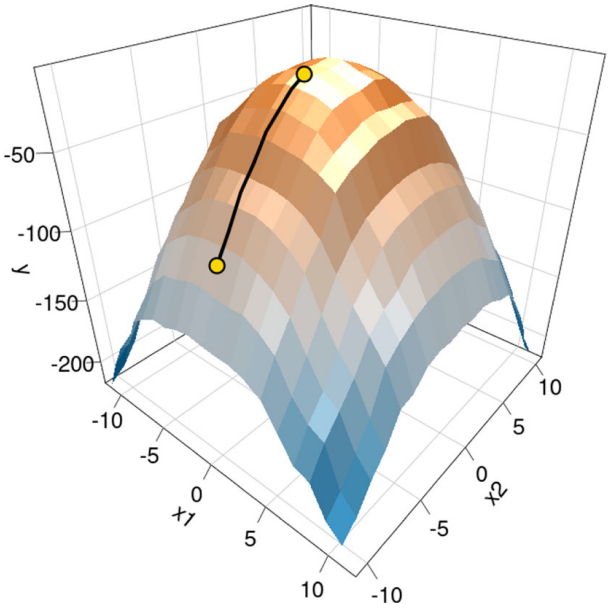


Fig. 1 The surface represents an exemplary prediction function dependent on two features. The FD can be considered a movement on the prediction function. We travel from point $(0, -9)$ to point $(0, -2)$ (Color figure online)

of the elements of S, or the vector $(x_j, \boldsymbol{x}_{-j})$ by $(x_1, \ldots, x_j, \ldots, x_p)$ although $j \in \{1, \ldots, p\}$. The i-th observed feature vector is denoted by $\boldsymbol{x}^{(i)}$ and corresponds to the target value $y^{(i)}$. We evaluate the prediction function with a set of training or test data $\mathscr{D} = \left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{n}$.

A finite difference (FD) of the prediction $\widehat{f}(\boldsymbol{x})$ w.r.t. x_j is defined as:

$$\text{FD}_{j,\boldsymbol{x},a,b} = \widehat{f}(x_1, \ldots, x_j + a, \ldots, x_p) - \widehat{f}(x_1, \ldots, x_j + b, \ldots, x_p)$$

The FD can be considered a movement on the prediction function (see Fig. 1). There are three common variants of FDs: forward ($a = h$, $b = 0$), backward ($a = 0$, $b = -h$), and central differences ($a = h, b = -h$). In the following, we only consider forward differences with $b = 0$ where the FD is denoted without b. Dividing the FD by $(a - b)$ corresponds to the difference quotient:

$$\frac{\text{FD}_{j,\boldsymbol{x},a,b}}{a - b} = \frac{\widehat{f}(x_1, \ldots, x_j + a, \ldots, x_p) - \widehat{f}(x_1, \ldots, x_j + b, \ldots, x_p)}{a - b}$$

The derivative is defined as the limit of the forward difference quotient when $a = h$ approaches zero:

$$\frac{\partial \widehat{f}(\boldsymbol{X})}{\partial X_j} \Big|_{\boldsymbol{X}=\boldsymbol{x}} = \lim_{h \to 0} \frac{\widehat{f}(x_1, \ldots, x_j + h, \ldots, x_p) - \widehat{f}(\boldsymbol{x})}{h}$$

We can numerically approximate the derivative with small values of h. For instance, we can use forward, backward, or symmetric FD quotients, which have varying error characteristics. As an example, consider a central FD quotient which is often used for derivative-based MEs (Leeper 2018):

$$\frac{\partial \widehat{f}(\boldsymbol{X})}{\partial X_j} \Big|_{\boldsymbol{X}=\boldsymbol{x}} \approx \frac{\widehat{f}(x_1, \ldots, x_j + h, \ldots, x_p) - \widehat{f}(x_1, \ldots, x_j - h, \ldots, x_p)}{2h} \quad , h > 0$$

3 Making sense of marginal effects

There is much ambiguity and confusion surrounding MEs. They are either defined in terms of derivatives or forward differences, and there is further confusion regarding the definition of categorical MEs.

3.1 Marginal effects for categorical features

MEs for categorical features are often computed as the change in prediction when the feature value changes from a reference category to another category (Williams 2012). In other words, for each observation, the observed categorical feature value is set to the reference category, and we record the change in prediction when changing it to every other category. Given k categories, this results in $k - 1$ MEs for each observation. Consider a categorical feature indexed by j with categories $C = \{c_1, \ldots, c_k\}$. We select a reference category $c_r \in C$. The categorical ME for an observation $\boldsymbol{x}$ and a single category $c_l \in C \backslash \{c_r\}$ corresponds to:

$$\text{ME}_{j,\boldsymbol{x},c_r,c_l} = \widehat{f}(c_l, \boldsymbol{x}_{-j}) - \widehat{f}(c_r, \boldsymbol{x}_{-j})$$

3.2 Marginal effects for continuous features

3.2.1 Definition as derivative

The most commonly used definition of MEs for continuous features corresponds to the derivative of the prediction function w.r.t. a feature. We will refer to this definition as the derivative ME (DME). In case of a linear prediction function, the interpretation of DMEs is simple: if the feature value increases by one unit, the prediction will increase by the DME estimate. Note that even the prediction function of a linear regression model can be non-linear if exponents of order ≥ 2 are included in the feature term. Similarly, in GLMs, the linear predictor is transformed (e.g., log-transformed in Poisson regression or logit-transformed in logistic regression).

3.2.2 Definition as forward difference

A distinct and often overlooked definition of MEs corresponds to the change in prediction with adjusted feature values, also referred to as discrete change (Mize et al. 2019) or difference in adjusted predictions (APs) (Williams 2012). This definition of MEs is based on a forward difference instead of a symmetric difference and does not require dividing the FD by the interval width. For this reason—and to establish a unified definition of MEs—we refer to this variant as the forward ME (FME):

$$\begin{aligned} \text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} &= \widehat{f}(x_1 + h_1, \ \ldots \ , x_s + h_s, \boldsymbol{x}_{-S}) - \widehat{f}(x_1, \ \ldots \ , x_s, \boldsymbol{x}_{-S}) \\ &= \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S}) - \widehat{f}(\boldsymbol{x}) \end{aligned} \tag{1}$$

A univariate FME for $h = 1$ is illustrated in Fig. 2. It corresponds to the change in prediction along the secant (orange, dotdashed) through the point of interest (prediction at x = 0.5) and the prediction at the feature value we receive after the feature change (x = 1.5).

Note that FMEs—as any other model-agnostic method—may result in model extrapolations if based on predictions in areas where the model was not trained with a sufficient amount of data. In Appendix A.2 , we discuss model extrapolations and how they relate to the computation of FMEs.

A technique that is subject to the additive recovery property only *recovers* terms of the prediction function that depend on the feature(s) of interest $\boldsymbol{x}_S$ or consist of interactions between the feature(s) of interest and other features, i.e., the method recovers no terms that exclusively depend on the remaining features $\boldsymbol{x}_{-S}$ (Apley and Zhu 2020). In Appendix B. 1, we derive the additive recovery property for FMEs.

3.2.3 Forward difference versus derivative

Note that we refer to using MEs to obtain **feature effect interpretations** (see Sect. 1), meaning changes in predicted outcome due to changes in feature values (locally and globally). In case of non-linear prediction functions, using DMEs for effect interpretations can lead to substantial misinterpretations (see Fig. 2). The slope of the tangent (green, dashed) at the point of interest (prediction at x = 0.5) corresponds to the DME. The default way to obtain a feature effect using the DME is to evaluate the tangent at the feature value we receive **after** changing feature values (in this case, we make a unit change, resulting in $x = 1.5$). This leads to substantial misinterpretations for non-linear prediction functions. In this case, there is an error (purple) almost as large as the actual change in prediction (the FME, blue). Although the computation of the DME does not require a step size, its interpretation does and is therefore error-prone. In contrast, the FME always indicates an exact change in prediction for any prediction function and is therefore much more interpretable. Only for linear prediction functions, the interpretation of both variants is equivalent.

There is a further advantage of FMEs over DMEs: derivatives are not suited to interpret piecewise constant prediction functions such as the ones created by tree-based algorithms. We discuss this point in more detail in Appendix A.3.

3.3 Variants and aggregations of marginal effects

There are three established variants or aggregations of MEs: The AME, MEM, and MER (Williams 2012), which can be computed for both DMEs and FMEs. In the following, we will use the notation of FMEs. Although we technically refer to the FAME, FMEM, and FMER, we omit the "forward" prefix in this case for reasons of simplicity:

(i) **Average marginal effect (AME):** The AME represents an estimate of the expected FME w.r.t. the distribution of $\boldsymbol{X}$. We estimate it via Monte-Carlo integration, i.e., we average the FMEs that were computed for each (randomly sampled) observation:

$$\mathbb{E}_{\boldsymbol{X}}\left[\text{FME}_{\boldsymbol{X},\boldsymbol{h}_S}\right] = \mathbb{E}_{\boldsymbol{X}}\left[\widehat{f}(\boldsymbol{X}_S + \boldsymbol{h}_S, \boldsymbol{X}_{-S}) - \widehat{f}(\boldsymbol{X})\right]$$

$$\text{AME}_{\mathscr{D},\boldsymbol{h}_S} = \frac{1}{n}\sum_{i=1}^{n}\left[\widehat{f}\left(\boldsymbol{x}_S{}^{(i)} + \boldsymbol{h}_S, \boldsymbol{x}_{-S}^{(i)}\right) - \widehat{f}\left(\boldsymbol{x}^{(i)}\right)\right]$$

(ii) **Marginal effect at means (MEM):** The MEM can be considered the reverse of the AME, i.e., it is the FME evaluated at the expectation of $\boldsymbol{X}$. We estimate the MEM by replacing all feature values with their sample distribution means:

$$\text{FME}_{\mathbb{E}_{\boldsymbol{X}}[\boldsymbol{X}],\boldsymbol{h}_S} = \widehat{f}\left(\mathbb{E}_{\boldsymbol{X}_S}[\boldsymbol{X}_S] + \boldsymbol{h}_S, \mathbb{E}_{\boldsymbol{X}_{-S}}\left[\boldsymbol{X}_{-S}\right]\right) - \widehat{f}\left(\mathbb{E}_{\boldsymbol{X}}[\boldsymbol{X}]\right)$$

$$\text{MEM}_{\mathscr{D},\boldsymbol{h}_S} = \widehat{f}\left(\left(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_S^{(i)}\right) + \boldsymbol{h}_S, \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}_{-S}^{(i)}\right) - \widehat{f}\left(\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{x}^{(i)}\right)$$

Note that averaging values is only sensible for continuous features. Williams (2012) defines a categorical MEM where all remaining features are set to their sample means (conditional on being continuous) and the feature of interest changes from a reference category to every other category.

(iii) **Marginal effect at representative values (MER):** Furthermore, we can replace specific feature values for all observations with manually specified values $\boldsymbol{x}^*$. It follows that the MEM is a special case of the MER where the specified values correspond to the sample means. MERs can be considered conditional FMEs, i.e., we compute FMEs while conditioning on certain feature values. The MER for a single observation with modified feature values $\boldsymbol{x}^*$ corresponds to:

$$\text{MER}_{\boldsymbol{x}^*,\boldsymbol{h}_S} = \widehat{f}\left(\boldsymbol{x}_S^* + \boldsymbol{h}_S, \boldsymbol{x}_{-S}^*\right) - \widehat{f}\left(\boldsymbol{x}^*\right)$$

The AME, MEM, and MER are mainly targeted at continuous features. In Sect. 4, we discuss computations for unified FMEs.

4 Model-agnostic forward marginal effects for arbitrary prediction functions

4.1 Unified definition of forward marginal effects

Note that both categorical MEs and FMEs are based on forward differences. We propose a unified definition of FMEs for continuous, categorical, and mixed-type features in S. Recall that the definition of FMEs for continuous features is given by Eq. (1):

$$\text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} = \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S}) - \widehat{f}(\boldsymbol{x}) \qquad \text{for continuous features } \boldsymbol{x}_S$$

We suggest an observation-specific categorical FME, where we first select a single category c_j and predict once with the observed value x_j and once where x_j has been

replaced by c_j:

$$\text{FME}_{\boldsymbol{x},c_j} = \widehat{f}(c_j, \boldsymbol{x}_{-j}) - \widehat{f}(\boldsymbol{x}) \qquad \text{for categorical } x_j$$

This definition of categorical FMEs is congruent with the definition of FMEs for continuous features, as we receive a single FME for a single observation with the observed feature value as the reference point. In other words, the reference category c_j for a categorical FME is conceptually identical to the step size h_j for a continuous FME. This implies that for observations where $x_j = c_j$, the categorical FME is zero. Continuous and categorical FMEs can be combined for mixed-data FMEs. Consider a set $S = \{j, l\}$ and the vector $\boldsymbol{h}_S = (h_j, c_l)$ with step size h_j for the j-th feature (which is continuous) and a category c_l for the l-th feature (which is categorical). A mixed-type FME is given by:

$$\text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} = \widehat{f}(x_j + h_j, c_l, \boldsymbol{x}_{-S}) - \widehat{f}(\boldsymbol{x}) \qquad \text{for continuous } x_j \text{ and categorical } x_l$$

We therefore remove any ambiguity from MEs through a unified definition and terminology based on forward differences for all feature types.

Categorical FMEs and the computation of MEMs and MERs: Categorical FMEs are also suited for computing a categorical AME. Note that we generally have less than n categorical FMEs different from zero, depending on the observed marginal distribution of $\{x_j^{(i)}\}_{i=1}^n$, which may affect the variance of the mean. Although the computation of MERs for categorical FMEs is possible, the MER obfuscates their interpretation by destroying the empirical distribution.

4.2 Non-linearity measure for continuous features

Although an FME represents the exact change in prediction and always accurately describes the movement on the prediction function, we lose information about the function's shape along the forward difference. It follows that when interpreting FMEs, we are at risk of misjudging the shape of the prediction function as a piecewise linear function. However, prediction functions created by ML algorithms are not only non-linear but also differ considerably in shape across the feature space. We suggest to augment the change in prediction with an NLM that quantifies the deviation between the prediction function and a linear reference function. First, the FME tells us the change in prediction for pre-specified changes in feature values. Then, the NLM tells us how accurately a linear effect resembles the change in prediction. The NLM thus represents a measure of confidence whether interpolations regarding the FME along the step are possible. For instance, assume the associated increase in a patient's diabetes risk is 5% for an increase in age by 10 years. The NLM tells us how confident we can be that aging the patient by 5 years will result in a 2.5% increase in risk.

4.2.1 Computation and interpretation

Linear reference function: A natural choice for the linear reference function is the secant intersecting both points of the forward difference (see Fig. 2). The secant for a

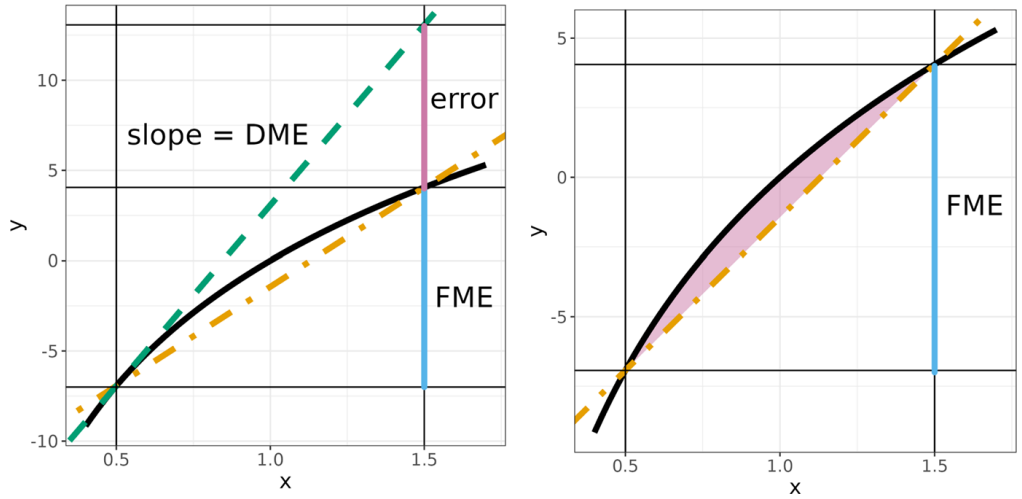


Fig. 2 Illustration of a univariate FME for $h = 1$ and a comparison to the corresponding DME. **Left**: The prediction function is black-colored. The DME is given by the slope of the tangent (green, dashed) at the point of interest (x = 0.5). The interpretation of the DME corresponds to the evaluation of the tangent value at x = 1.5, which is subject to an error (purple) almost as large as the actual change in prediction. The FME (blue) equals the change in prediction along the secant (orange, dotdashed) through the prediction at x = 0.5 and at x = 1.5. **Right**: The deviation between the prediction function (black) and linear secant (orange, dotdashed) can be quantified via the purple area. For the NLM, we put this integral in relation to the integral of the area between the prediction function and the mean prediction (Color figure online)

multivariate FME corresponds to:

$$g_{\boldsymbol{x},\boldsymbol{h}_S}(t) = \begin{pmatrix} x_1 + t \cdot h_1 \\ \vdots \\ x_s + t \cdot h_s \\ \vdots \\ x_p \\ \widehat{f}(\boldsymbol{x}) + t \cdot \text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} \end{pmatrix}$$

The multivariate secant considers equally proportional changes in all features. Figure 3 visualizes the discrepancy between the prediction function and the secant along a two-dimensional FME. If the NLM indicates linearity, we can infer that if *all* individual feature changes are multiplied by a scalar $t \in [0, 1]$, the FME would change by t as well.

Definition of the NLM: Comparing the prediction function against the linear reference function along the FME requires a normalized metric that indicates the degree of similarity between functions or sets of points. Established metrics in geometry include the Hausdorff (Belogay et al. 1997) and Fréchet (Alt and Godau 1995) distances. Another option is to integrate the absolute or squared deviation between both functions. These approaches have the common disadvantage of not being normalized, i.e., the degree of non-linearity is scale-dependent.
Molnar et al. (2020) compare non-linear function segments against linear models via the coefficient of determination R^2. In this case, R^2 indicates how well the linear

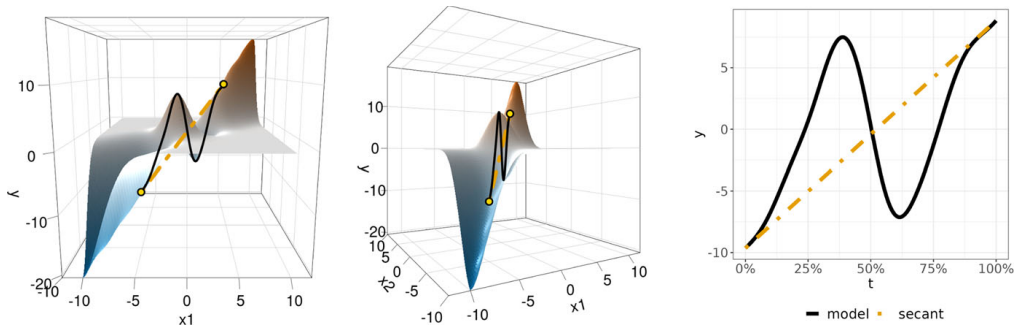


Fig. 3 A non-linear prediction function, the path along its surface, and the corresponding secant along a two-dimensional FME from point $(-5, -5)$ to point $(5, 5)$. The right plot depicts the parameterization in terms of t as the percentage of the step size $\boldsymbol{h}_S$. This type of parameterization and visualization is possible for any dimensionality of $\boldsymbol{h}_S$

reference function is able to explain the non-linear prediction function compared to the most uninformative baseline model, i.e., one that always predicts the prediction function through its mean value. As we do not have observed data points along the forward difference, points would need to be obtained through (Quasi-)Monte-Carlo sampling, whose error rates heavily depend on the number of sampled points. As both the FME and the linear reference function are evaluated along the same single path across the feature space, their deviation can be formulated as a line integral. Hence, we are able to extend the concept of R^2 to continuous integrals, comparing the integral of the squared deviation between the prediction function and the secant, and the integral of the squared deviation between the prediction function and its mean value. The line integral is univariate and can be numerically approximated with various techniques such as Gaussian quadrature.

The parametrization of the path through the feature space is given by $\gamma : [0, 1] \mapsto \mathscr{X}$, where $\gamma(0) = \boldsymbol{x}$ and $\gamma(1) = (\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S})$. The line integral of the squared deviation between prediction function and secant along the forward difference corresponds to:

$$(\mathrm{I}) = \int_0^1 \left(\widehat{f}(\gamma(t)) - g_{\boldsymbol{x},\boldsymbol{h}_S}(t)\right)^2 \left|\left|\frac{\partial \gamma(t)}{\partial t}\right|\right|_2 dt$$

with

$$\gamma(t) = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} + t \cdot \begin{pmatrix} h_1 \\ \vdots \\ h_s \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad t \in [0, 1]$$

and

$$\left|\left|\frac{\partial \gamma(t)}{\partial t}\right|\right|_2 = \sqrt{h_1^2 + \cdots + h_s^2}$$

The integral of the squared deviation between the prediction function and the mean prediction is used as a baseline. The mean prediction is given by the integral of the prediction function along the forward difference, divided by the length of the path:

$$\begin{aligned}
\overline{\widehat{f}(t)} &= \frac{\int_0^1 \widehat{f}(\gamma(t)) \left|\left|\frac{\partial \gamma(t)}{\partial t}\right|\right|_2 dt}{\int_0^1 \left|\left|\frac{\partial \gamma(t)}{\partial t}\right|\right|_2 dt} \\
&= \int_0^1 \widehat{f}(\gamma(t))\, dt \\
\text{(II)} &= \int_0^1 \left(\widehat{f}(\gamma(t)) - \overline{\widehat{f}(t)}\right)^2 \left|\left|\frac{\partial \gamma(t)}{\partial t}\right|\right|_2 dt
\end{aligned}$$

The $\text{NLM}_{\boldsymbol{x},\boldsymbol{h}_S}$ is defined as:

$$\text{NLM}_{\boldsymbol{x},\boldsymbol{h}_S} = 1 - \frac{\text{(I)}}{\text{(II)}}$$

Interpretation: The NLM has an upper limit of 1 and indicates how well the secant can explain the prediction function, compared to the baseline model of using the mean prediction. For a value of 1, the prediction function is equivalent to the secant (perfect linearity). A lower value indicates increasing non-linearity of the prediction function. For negative values, the mean prediction better predicts values on the prediction function than the secant (severe non-linearity). We suggest to use 0 as a hard bound to indicate non-linearity and values on the interval]0, 1[as an optional soft bound.
Advantages of the NLM: Given only univariate changes in feature values, we may visually assess the non-linearity of the feature effect with an ICE curve (see Sect. 5). However, the NLM quantifies non-linearity in a single metric. For one, this facilitates interpretations: for instance, in Fig. 13, the average NLM correctly diagnoses linear effects of the features x_4 and x_5 in Friedman's regression problem. Second, this information can be further utilized in an informative summary output of the prediction function: in Sect. 4.3, we estimate feature effects for population subgroups where individual NLM values can be averaged to describe average non-linearities within subgroups. For bivariate feature changes, the NLM greatly simplifies non-linearity assessments: as an example, consider Fig. 12 where the sinus curve's point of inflection for the interaction of x_1 and x_2 in Friedman's regression problem can be detected with NLM values. Lastly, given changes in more than two features, visual interpretation techniques such as the ICE and PD are not applicable at all. As opposed to this, the NLM is defined in arbitrary dimensions and can be used for feature changes of any dimensionality (see Fig. 20 for an example with a trivariate feature change).

4.2.2 Selecting step sizes and linear trust region

The step size is determined both by the question that is being addressed and the scale of the feature at training time. In many cases, an interpretable or intuitive step size is preferable. For instance, body weight tends to be expressed in kilograms, thus making 1 kg (as opposed to 1 g) a natural increment. Contextual information, too, dictates step sizes. For instance, a 1 kg difference in body weight might not elicit many physiological changes. One might suspect, for instance, a 5 kg difference to elicit noticeable changes and to provide an actionable model interpretation, where the patient can be advised to lose weight if the model predicts a favorable outcome of that action. If a natural unit or contextual information is not available, the units recorded in the data set make a reasonable default step size. This also links back to the natural interpretation of LMs, whose beta coefficients indicate the change in predicted outcome due to a unit change in the feature value.

Dispersion-based step sizes: Without contextual information, dispersion-based measures such as one standard deviation can also be used as step sizes (Mize et al. 2019). Other options include, e.g., percentages of the interquartile range (IQR) or the mean / median absolute deviation. Furthermore, we can compute and visualize FME and NLM distributions for various step sizes or step size combinations for multivariate FMEs (see Fig. 18 for an example).

Local linear trust region (LLTR): In selected applications it might be of interest to have confidence in the linearity of FMEs, which can be ensured with an NLM threshold. Figure 4 visualizes an example by Molnar (2022) where LIME (see Sect. 5) fails to accurately explain the black box prediction for a data point depending on the chosen kernel width. We wish to explain the predictions of the black box model (black line) for a single data point (black dot). For kernel widths 0.75 or 2, the local surrogate indicates no or a positive effect of x on the predicted target, while the actual effect is negative. In contrast, the FME can be used to compute the exact feature effect where the NLM provides an LLTR (visualized by the orange arrows). In this example, traversing the black box model from the black dot along each arrow is associated with an NLM ≥ 0.9, i.e., an approximately linear FME. Which NLM threshold indicates linearity is debatable. For this paper, we choose a very high threshold of 0.9 to leave a margin of safety. The right plot visualizes FME and NLM pairs for each step of the LLTR. Steps that cannot be included in the LLTR are greyed out. An LLTR for multivariate steps is visualized in Fig. 17.

Step sizes and model extrapolations: The step size cannot vary indefinitely without risking model extrapolations. Furthermore, when using non-training data or training data in low-density regions to compute FMEs, we are at risk of the model extrapolating without actually traversing the feature space. If the points $\boldsymbol{x}$ or $(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S})$ are classified as extrapolation points (EPs), the FME should be interpreted with caution or the observation be excluded from the analysis.

Fig. 5 demonstrates the perils of model extrapolations when using FMEs. We draw points of a single feature x from a uniform distribution on the interval $[-5, 5]$. The target is generated as $y = x^2 + \epsilon$, where ϵ is drawn from $N(0, 1)$. A random forest is trained to predict y given x. All points $x \notin [-5, 5]$ are located outside the range of the training data and can be considered EPs. We compute FMEs with a step size of 1.

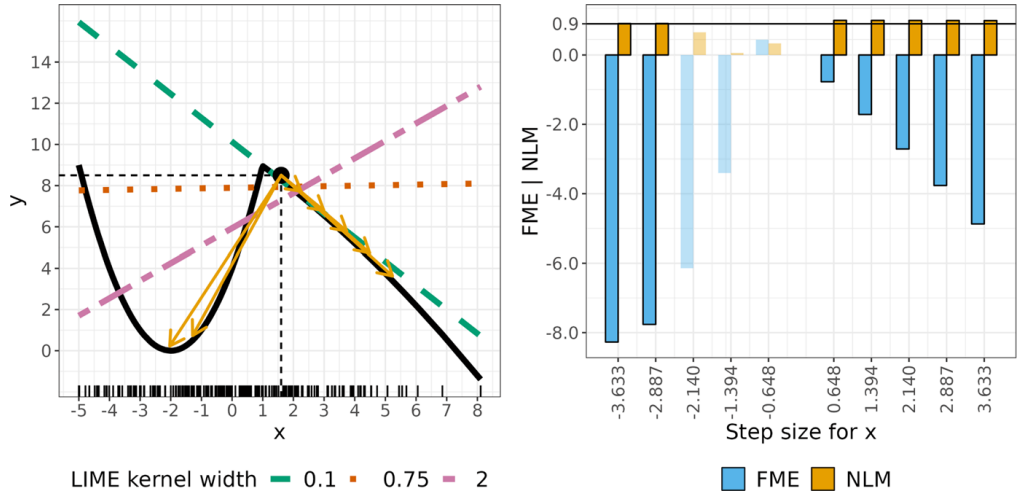


Fig. 4 **Left:** Explaining a single local prediction at the black dot (x = 1.6, $\widehat{f}(x) = 8.5$). The black box model predictions are given by the black line. Local surrogate explanations via LIME differ considerably depending on the chosen kernel width (straight lines, kernel width indicated by shape and color). In contrast, the FME always represents an exact forward difference between the black dot and points on the prediction function (where the secant is visualized by the orange arrows). The step sizes associated with the arrows represent an exemplary LLTR of FMEs for which the NLM ≥ 0.9 (approximate linearity). **Right:** Visualization of LLTR with pairs of FME and NLM for each explored step. Step sizes with an NLM < 0.9 are greyed out (Color figure online)

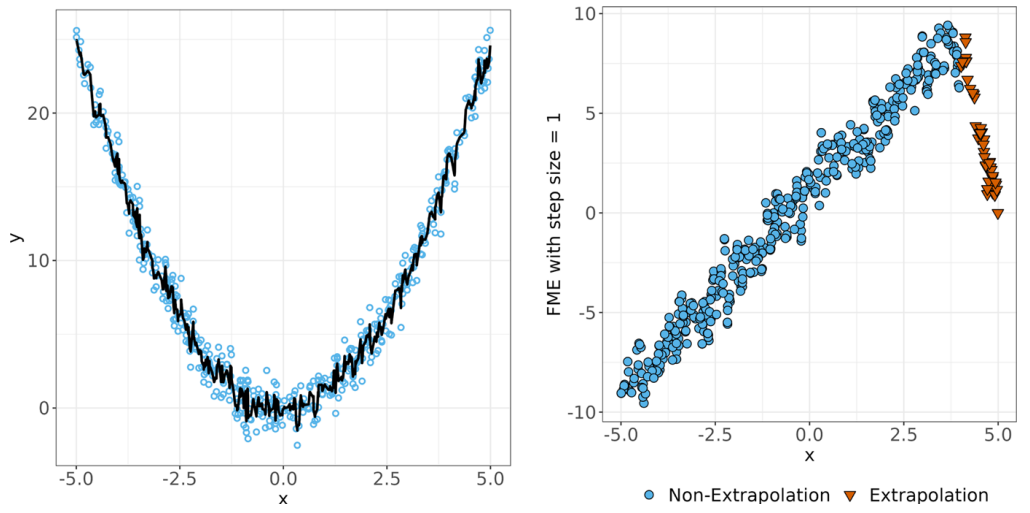


Fig. 5 **Left**: A random forest is trained on a single feature x with a quadratic effect on the target. The training space corresponds to the interval $[-5, 5]$. **Right**: We compute an FME with a step size of 1 for each observation. After moving 1 unit in x direction, points with $x > 4$ are considered EPs (red triangles). The random forest extrapolates and predicts unreliably in this area of the feature space. The resulting FMEs are irregular and should not be used for interpretation purposes (Color figure online)

By implication, all FMEs with $x > 4$ are based on model extrapolations. FMEs based on model extrapolations exhibit a considerably different pattern and should not be used for interpretation purposes, as they convey an incorrect impression of the feature effect of x.

4.3 Regional feature effects with conditional average marginal effects

It is desirable to summarize the feature effect in a single metric, similarly to the parameter-focused interpretation of LMs. For instance, one is often interested in the expected FME (for the entire feature space), which can be estimated via the AME. However, averaging heterogeneous FMEs to the AME is not globally representative of non-linear prediction functions such as the ones created by ML algorithms. A heterogeneous distribution of FMEs requires a more local evaluation. As opposed to conditioning on feature values in the case of MERs (local), we further suggest to condition on specific feature subspaces (regional). The cAME is an estimate of the expected FME for the random vector $\boldsymbol{X}_{[\,]}$ with a restricted sample space $\mathscr{X}_{[\,]}$. It is computed for a subsample of observations $\mathscr{D}_{[\,]}$ sampled from $\mathscr{X}_{[\,]}$:

$$\begin{aligned} \text{cAME}_{\mathscr{D}_{[]},\boldsymbol{h}_S} &= \mathbb{E}_{\boldsymbol{X}_{[]}}\left[\widehat{\text{FME}}_{\boldsymbol{X}_{[]},\boldsymbol{h}_S}\right] \\ &= \frac{1}{n_{[]}} \sum_{i:\boldsymbol{x}^{(i)} \in \mathscr{D}_{[]}} \left[\widehat{f}\left(\boldsymbol{x}_S^{(i)} + \boldsymbol{h}_S, \boldsymbol{x}_{-S}^{(i)}\right) - \widehat{f}\left(\boldsymbol{x}^{(i)}\right)\right] \\ &\text{with} \quad n_{[]} = |\mathscr{D}_{[]}| \end{aligned} \tag{2}$$

A population subgroup $\mathscr{X}_{[\,]}$ corresponds to a subspace of the feature space $\mathscr{X}$, e.g., a range of health characteristics of patients with a certain predisposition of developing a disease. The subsample $\mathscr{D}_{[\,]}$ consists of data that were drawn from this subspace, e.g., patients with said predisposition that partook in a study. Note that in our case, we are looking for subgroups with homogeneous effects on the model prediction, e.g., patients for whom increasing their age has similar effects on the predicted disease risk. Even though such population subgroups might exist (in many cases they may not), the model fit fundamentally determines whether we can find subgroups with homogeneous effects for the trained model.

4.3.1 Desiderata for finding subgroups

Note that Eq. (2) is defined in general terms, conditional on an arbitrary subspace $\mathscr{X}_{[\,]}$. We can arbitrarily partition the feature space, determine corresponding subsets of observed data, and run the estimator in Eq. (2) for each subsample to estimate expected conditional FMEs. However, recall that our goal is to find accurate descriptors of feature effects for the trained model across the feature space. Therefore, we formulate multiple desiderata for these subspaces and the corresponding subsamples (hereafter referred to as subgroups):

- **Within-group effect homogeneity:** FME variance inside subgroups shall be minimized.
- **Between-group effect heterogeneity:** cAMEs of subgroups shall be heterogeneous.
- **Full segmentation:** The data shall be fully segmented into subgroups.
- **Non-congruence:** Subgroups shall not overlap with each other.
- **Confidence:** Larger subgroups are preferred over smaller subgroups.

- **Stability:** Subgroups shall be stable w.r.t. variations in the data.

Evidently, certain desiderata are difficult to meet. For instance, we can strive to minimize FME variance within a single subgroup, but this might increase FME variance within other subgroups.
Note that the philosophy of regional or semi-global feature effects somewhat deviates from our previous philosophy of obtaining simple and stable local model explanations. Finding subgroups with more homogeneous local explanations by modeling FME patterns necessitates some sort of approximation. In the following section, we model FME patterns with decision trees and discuss the upsides and downsides of this approach.

4.3.2 Estimation using decision trees

Decision tree learning is an ideal scheme to partition the entire feature space into mutually exclusive subspaces, thus finding population subgroups. Growing a tree by global optimization poses considerable computational difficulties and corresponds to an NP-complete problem (Norouzi et al. 2015). Recent developments in computer science and engineering can be explored to revisit global decision tree optimization from a different perspective, e.g., Bertsimas and Dunn (2017) explore mixed-integer optimization to find globally optimal decision trees. To reduce computational complexity, the established way (which is also commonly available in many software implementations) is through recursive partitioning (RP), optimizing an objective function in a greedy way for each tree node.
Over the last decades, a large variety of RP methods has been proposed (Loh 2014), with no gold standard having crystallized to date. In principle, any RP method that is able to process continuous targets can be used to find subgroups, e.g., classification and regression trees (CART) (Breiman et al. 1984; Hastie et al. 2001), which is one of the most popular approaches. Trees have been demonstrated to be notoriously unstable w.r.t. perturbations in input data (Zhou et al. 2023; Last et al. 2002). Tree ensembles, such as random forests (Breiman 2001a), reduce variance but lose interpretability as a single tree structure. Exchanging splits along a single path results in structurally different but logically equivalent trees (Turney 1995). It follows that two structurally very distinct trees can create the same or similar subspaces. We are therefore not interested in the structure of the tree itself, but in the subgroups it induces.
Stabilization of RP: As formulated earlier, we strive to find subgroups that are stable w.r.t. variations in the data. For RP, one should therefore strive to stabilize splits. A branch of RP methods incorporates statistical theory into the split procedure. Variants include conditional inference trees (CTREE) (Hothorn et al. 2006), which use a permutation test to find statistically significant splits; model-based recursive partitioning (MOB) (Zeileis et al. 2008), which fits node models and tests the instability of the model parameters w.r.t. partitioning the data; or approximation trees (Zhou et al. 2023), which generate artificially created samples for significance testing of tree splits. Seibold et al. (2016) use MOB to find patient subgroups with similar treatment effects in a medical context. Furthermore, we can assess the stability of feature and split point selection for arbitrary tree models by resampling the training data and retraining the tree (Philipp et al. 2016). The variance and instability of decision trees partly

stems from binary splits, as a decision higher up cascades through the entire tree and results in different splits lower down the tree (Hastie et al. 2001). Using multiway trees, which also partition the entire feature space, would therefore improve stability. However, multiway splits are associated with a considerable increase in computational complexity and are therefore often discarded in favor of binary splitting (Zeileis et al. 2008). For the remainder of the paper, we use CTREE to find subgroups and compute cAMEs.

4.3.3 Confidence intervals for the cAME and cANLM

Given estimates of the expected conditional FME, it is desirable to estimate the expected conditional NLM for the corresponding subspaces as well. Analogously to the AME, we can compute an average NLM (ANLM) by globally averaging NLMs and a conditional ANLM (cANLM) by averaging NLMs within a subgroup. The cANLM gives us an estimate of the expected non-linearity of the prediction function for the given movements along the feature space, conditional on a feature subspace.
A lower standard deviation (SD) of FMEs and NLM values increases confidence in our estimates and vice versa, and a larger number of observations increases confidence in our estimates and vice versa. Although we do not specify a distribution of the underlying FMEs or NLMs, constructing a confidence interval (CI) is possible via the central limit theorem. As the cAME and cANLM are sample averages of all FMEs and NLMs for each subgroup, we can construct a t-statistic (as the SD is estimated) for large sample sizes. Given a subgroup $\mathscr{D}_{[\,]}$ that contains $n_{[\,]}$ observations, mean ($\text{cAME}_{\mathscr{D}_{[\,]},\boldsymbol{h}_S}$ and $\text{cANLM}_{\mathscr{D}_{[\,]},\boldsymbol{h}_S}$) and SD ($\text{SD}_{\text{FME},\,[\,]}$ and $\text{SD}_{\text{NLM},\,[\,]}$) values, the confidence level α, and the values of the t-statistic with $n_{[\,]} - 1$ degrees of freedom at the $1 - \frac{\alpha}{2}$ percentile ($t_{1-\frac{\alpha}{2},\, n_{[\,]}-1}$), the CIs correspond to:

$$\text{CI}_{\text{cAME},\,1-\alpha} = \left[\text{cAME}_{\mathscr{D}_{[\,]},\boldsymbol{h}_S} - t_{1-\frac{\alpha}{2},\, n_{[\,]}-1}\frac{\text{SD}_{\text{FME},\,[\,]}}{\sqrt{n_{[\,]}}}\,,\right.$$
$$\left.\text{cAME}_{\mathscr{D}_{[\,]},\boldsymbol{h}_S} + t_{1-\frac{\alpha}{2},\, n_{[\,]}-1}\frac{\text{SD}_{\text{FME},\,[\,]}}{\sqrt{n_{[\,]}}}\right]$$
$$\text{CI}_{\text{cANLM},\,1-\alpha} = \left[\text{cANLM}_{\mathscr{D}_{[\,]},\boldsymbol{h}_S} - t_{1-\frac{\alpha}{2},\, n_{[\,]}-1}\frac{\text{SD}_{\text{NLM},\,[\,]}}{\sqrt{n_{[\,]}}}\,,\right.$$
$$\left.\text{cANLM}_{\mathscr{D}_{[\,]},\boldsymbol{h}_S} + t_{1-\frac{\alpha}{2},\, n_{[\,]}-1}\frac{\text{SD}_{\text{NLM},\,[\,]}}{\sqrt{n_{[\,]}}}\right]$$

One option to ensure that the lower sample size threshold for CIs is valid is to specify a minimum size for each subgroup, e.g., in the case of RP, not growing the tree too large.

5 Related work

5.1 Statistics and applied fields

MEs have been discussed extensively in the literature on statistics and statistical software, e.g., by Ai and Norton (2003), Greene (2012), Norton et al. (2019), or Mullahy (2017). The `margins` command is a part of Stata (StataCorp. 2023) and was originally implemented by Bartus (2005). A brief description of the `margins` command is given by Williams (2012). Leeper (2018) provides an overview on DMEs and their variations as well as a port of Stata's functionality to R. The R package `marginaleffects` (Arel-Bundock 2023) supports various variants of MEs including FMEs. Ramsey and Bergtold (2021) compute an ME for a single-hidden-layer feed-forward backpropagation artificial neural network by demonstrating its interpretation is equivalent to a logistic regression model with a flexible index function. Zhao et al. (2020) apply model-agnostic DMEs to ML models in the context of analyzing travel behavior. Furthermore, they mention the unsuitability of derivatives for tree-based prediction functions such as random forests.
Mize et al. (2019) provide a test framework for cross-model differences of MEs. They refer to an ME based on a forward difference as a discrete change and to the corresponding AMEs as average discrete changes. Gelman and Pardoe (2007) propose the predictive effect as a local feature effect measure. The predictive effect is a univariate forward difference, divided by the change in feature values (i.e., the step size). This differentiates it from the FME which is also defined for multivariate feature changes and which is not divided by the step size, i.e., it provides a change in prediction as opposed to a rate of change. Furthermore, the authors propose an average predictive effect that corresponds to the average of multiple predictive effects that were measured at distinct feature values and model parameters. It is a generalization of the AME that may be estimated with artificially created data points (as opposed to the sample at hand) and incorporates model comparisons (measured with different model parameters).

5.2 Interpretable machine learning

The most commonly used techniques to determine feature effects include the individual conditional expectation (ICE) (Goldstein et al. 2015), the partial dependence (PD) (Friedman 2001), accumulated local effects (ALE) (Apley and Zhu 2020), Shapley values (Štrumbelj and Kononenko 2014), Shapley additive explanations (SHAP) (Lundberg and Lee 2017), local interpretable model-agnostic explanations (LIME) (Ribeiro et al. 2016), and counterfactual explanations (Wachter et al. 2018). Counterfactual explanations indicate the smallest necessary change in feature values to receive the desired prediction and represent the counterpart to MEs. Goldstein et al. (2015) propose derivative ICE (d-ICE) plots to detect interactions. The d-ICE is a univariate ICE where the numeric derivative w.r.t. the feature of interest is computed pointwise after a smoothing procedure. Symbolic derivatives are commonly used to determine the importance of features for neural networks (Ancona et al. 2018). While FMEs provide

interpretations in terms of prediction *changes*, most methods provide an interpretation in terms of prediction *levels*. LIME is an alternative option that returns interpretable parameters (i.e., rates of change in prediction) of a local surrogate model. LIME, and to a lesser extent SHAP, have been demonstrated to provide unreliable interpretations in some cases (Slack et al. 2020). Furthermore, many techniques in IML are interpreted visually (e.g., ICEs, the PD, ALE plots) and are therefore limited to feature value changes in at most two dimensions. FMEs are not limited by the dimension of the intervention in feature values, as any change in feature values—regardless of its dimensionality—always results in a single FME.

5.2.1 Relation between forward marginal effects, the individual conditional expectation, and partial dependence

Given a data point $\boldsymbol{x}$, the ICE of a feature set S corresponds to the prediction as a function of substituted values $\boldsymbol{x}_S^*$ where $\boldsymbol{x}_{-S}$ is kept constant:

$$\text{ICE}_{\boldsymbol{x},S}(\boldsymbol{x}_S^*) = \widehat{f}(\boldsymbol{x}_S^*, \boldsymbol{x}_{-S})$$

The PD on a feature set S corresponds to the expectation of $\widehat{f}(\boldsymbol{X})$ w.r.t. the marginal distribution of $\boldsymbol{X}_{-S}$. It is estimated via Monte-Carlo integration where the draws $\boldsymbol{x}_{-S}$ correspond to the sample values:

$$\widehat{\text{PD}}_{\mathscr{D},S}(\boldsymbol{x}_S) = \frac{1}{n} \sum_{i=1}^{n} \widehat{f}\left(\boldsymbol{x}_S, \boldsymbol{x}_{-S}^{(i)}\right)$$

We can visually demonstrate that in the univariate case, the FME is equivalent to the vertical difference between two points on an ICE curve. However, the AME is only equivalent to the vertical difference between two points on the PD curve for linear prediction functions (see Fig. 6). We generalize this result to the multivariate FME and ICE, as well as the multivariate forward AME and PD (see Theorem 3 and Theorem 4 in Appendix B.2). Visually assessing changes in prediction due to a change in feature values is difficult to impossible in more than two dimensions. High-dimensional feature value changes therefore pose a natural advantage for FMEs over techniques such as the ICE, PD, or ALE, which are mainly interpreted visually.

5.2.2 Comparison to LIME

LIME—one of the most popular model-agnostic feature effect methods—resembles the interpretation given by an FME. It also serves as a local technique, explaining the model for a single observation. LIME samples instances, predicts, and weights the predictions by the instances' proximity to the instance of interest using a kernel function. Afterwards, an interpretable surrogate model is trained on the weighted predictions. The authors choose a sparse linear model, whose beta coefficients provide an interpretation similar to the FME.

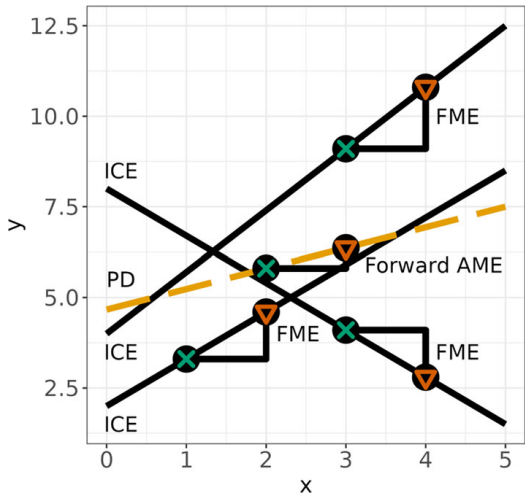


Fig. 6 Three ICE curves are black-colored. The PD is the average of all ICE curves (orange, dashed). For each ICE curve, we have a single observation, visualized by the corresponding green x-shaped point. We compute the FME at each observation with a step size of 1, which results in the corresponding red triangle-shaped point. The FMEs are equivalent to the vertical difference between two points on the ICE curves. If the prediction function is linear in the feature of interest, the average of all FMEs is equivalent to the vertical difference between two points on the PD (Color figure online)

But there is a fundamental difference between both approaches. The FME directly works on the prediction function, while LIME trains a local surrogate model. The latter is therefore affected by an additional layer of complexity and uncertainty. The authors suggest to use LASSO regression, which requires choosing a regularization constant. Furthermore, one must select a similarity kernel defined on a distance function with a width parameter which has tremendous effects on the resulting model explanation (see Fig. 4 for an example). The model interpretation is therefore fundamentally determined by multiple parameters. Furthermore, certain surrogate models are incapable of explaining certain model behaviors and may potentially mislead the practitioner to believe the interpretation (Ribeiro et al. 2016). A linear surrogate model may not be able to describe extreme non-linearities of the prediction function, even within a single locality of the feature space. In contrast, the only parameters for the FME are the features and the step sizes. Without question, the choice of parameters for FMEs also significantly affects the interpretation. However, we argue that their impact is much clearer than in LIME, e.g., a change in a feature such as age is much more meaningful than a different width parameter in LIME. In fact, we argue that the motivation behind both approaches is fundamentally different. For FMEs, we start with a meaningful interpretation concept in mind, e.g., we may be interested in the combined effects of increasing age and weight on the disease risk. For LIME, we start with a single observation, trying to distill the black box model behavior within this specific locality into a surrogate model.

In addition to the sensitivity of results regarding parameter choices, LIME is notoriously unstable even with fixed parameters. Zhou et al. (2021) note that repeated runs using the same explanation algorithm on the same model for the same observa-

tion results in different model explanations, and they suggest significance testing as a remedy. In contrast, FMEs with fixed parameters are deterministic.
As noted above, the authors of LIME mention that the faithfulness of the local surrogate may be diminished by extreme non-linearities of the model, even within the locality of the instance of interest. This exact same critique holds for the FME (see Sect. 4.2). Hence, we introduce the NLM, which essentially corresponds to a measure of faithfulness of the FME and whose concept can potentially be used for other methods as well. One could also use the coefficient of determination R^2 to measure the goodness-of-fit of the linear surrogate to the pseudo sample in LIME. However, we argue that the goodness-of-fit to a highly uncertain pseudo sample is a questionable way of measuring an explanation's faithfulness.
Furthermore, the authors of LIME note that insights into the global workings of the model may be gained by evaluating multiple local explanations. As there usually are time constraints so that not all instances can be evaluated, an algorithm suggests a subset of representative instances. Although this approach avoids the issue of misrepresenting global effects by averaging local explanations, it also misses the opportunity to provide meaningful regional explanations. This is where the cAME comes into play. It is motivated by the goal to aggregate local interpretations while staying faithful to the underlying predictive model. Note that a subset of representative instances—as suggested by Ribeiro et al. (2016)—can also be used to compute representative FMEs.

5.3 Sensitivity analysis

The goal of sensitivity analysis (SA) is to determine how uncertainty in the model output can be attributed to uncertainty in the model input, i.e., determining the importance of input variables (Saltelli et al. 2008). Techniques based on FDs are common in SA (Razavi et al. 2021). The numeric derivative of the function to be evaluated w.r.t. an input variable serves as the natural definition of local importance in SA. The elementary effect (EE) was first introduced as part of the Morris method (Morris 1991) as a screening tool for important inputs. The EE corresponds to a univariate forward difference quotient with variable step sizes, i.e., it is a generalization of the derivative. Variogram-based methods analyze forward differences computed at numerous pairs of points across the feature space (Razavi and Gupta 2016). Derivative-based global sensitivity measures (Sobol and Kucherenko 2010) provide a global feature importance metric by averaging derivatives at points obtained via random or quasi-random sampling.

6 Simulations

Here, we present multiple simulation scenarios to highlight the workings and interplay of FMEs, the NLM, and the cAME. In all following sections, we use Simpson's 3/8 rule for the computation of the NLM and CTREE to find subgroups.

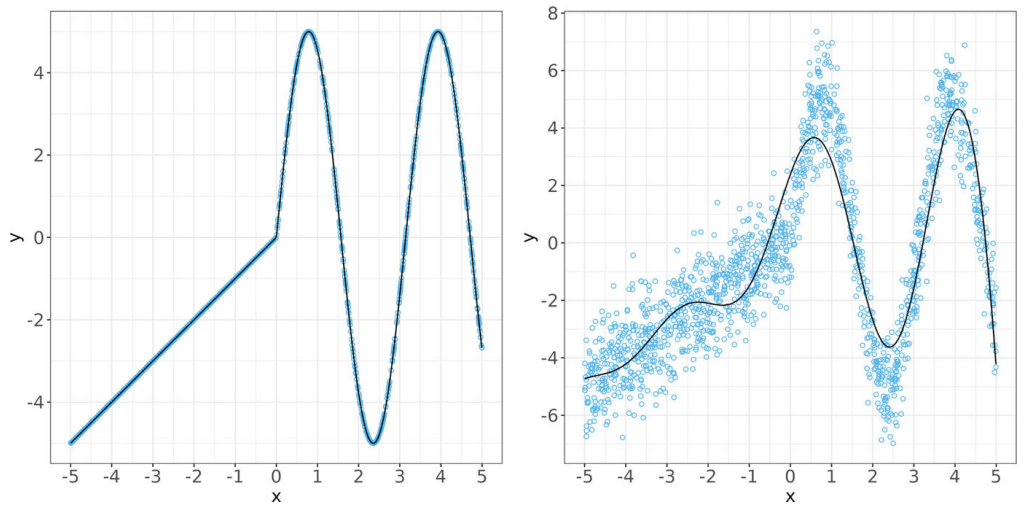


Fig. 7 The target is determined by a single feature x. On the interval $[-5, 0[$ there is a linear feature effect. On the interval $[0, 5]$ the functional relationship consists of a transformed sine wave. We first use the DGP, then add random noise on top of the data and train an SVM (Color figure online)

6.1 Univariate data without noise

We start with a univariate scenario without random noise and work directly with the data generating process (DGP). This way, we can evaluate how introducing noise affects the information gained from FMEs in the subsequent simulation. We simulate a single feature x, uniformly distributed on $[-5, 5]$, and define f as:

$$f(x) = \begin{cases} x & x < 0 \\ 5\sin(2x) & x \geq 0 \end{cases}$$

The data are visualized in Fig. 7. An FME with step size $h = 2$ is computed for each observation. We use CTREE on the FMEs to find subgroups. Subsequently, all observations' NLM values are averaged to cANLM values on the subspaces of the cAMEs. Our computations are visualized in Fig. 8. Vertical lines indicate tree splits, and corresponding FME or NLM subgroup averages are indicated by horizontal lines. In the univariate case, we see a direct relationship between the shape of the DGP and the FMEs and NLM values. The NLM has ramifications on the interpretation of the FMEs. For instance, for $x = -3$, increasing x by 2 units increases the predicted target value by 2 units, and we can conclude that the same holds proportionally for feature value changes of smaller magnitudes, e.g., a change of 1 unit results in an FME of 1, etc. On the contrary, given an observation $x = 1$, the NLM indicates considerable non-linearity. For this observation, we cannot draw conclusions about FMEs with smaller step sizes than 2 units.

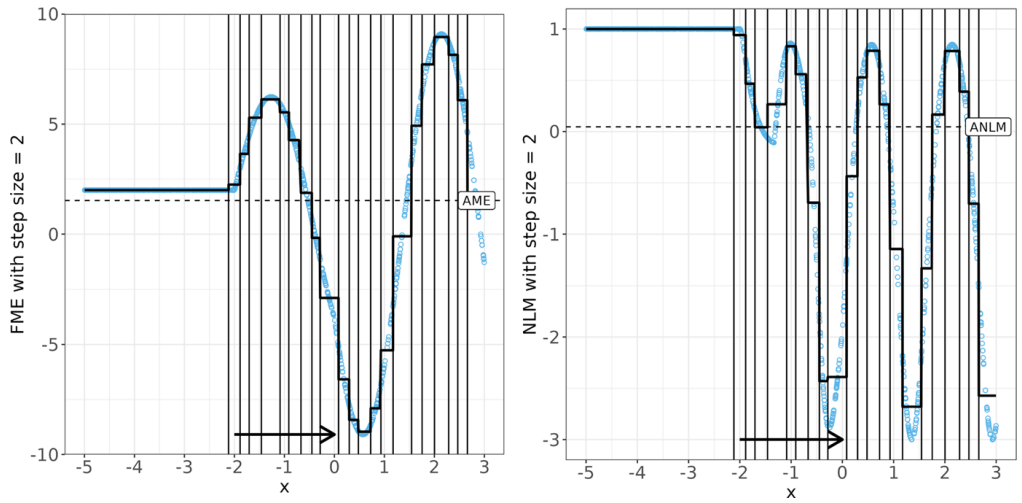


Fig. 8 Univariate data without noise. For each point, moving in x direction by the length of the arrow results in the FME / NLM indicated on the vertical axis. **Left**: FMEs with step size $h = 2$. A regression tree partitions the feature space into subspaces (in this case intervals) where the FMEs are most homogeneous. The horizontal lines correspond to the cAMEs. **Right**: NLM values and cANLMs for each subspace (Color figure online)

6.2 Univariate data with noise

We proceed to add random noise $\epsilon \sim N(0, 1)$ on top of the data and tune the regularization and sigma parameters of a support vector machine (SVM) with a radial basis function kernel (see Fig. 7). As we now employ a predictive model, we must avoid potential model extrapolations. The forward location of all points with $x > 3$ falls outside the range of the training data. After removing all extrapolation points, we evaluate the FMEs and NLMs of all observations with $x \in [-5, 3]$ (see Fig. 9). In this case, we can visually assess that the predictions of the SVM resemble the DGP but also factor in noise (see Fig. 7). e.g., the SVM prediction function is non-linear in linear regions of the DGP, which affects the FMEs and NLMs. This demonstrates that FMEs can only be used to explain the DGP if the model describes it accurately.

6.3 Bivariate data with univariate feature change

We next augment the univariate data with one additional feature in order to empirically evaluate the additive recovery property of the FME (see Appendix B.1). Due to potential model extrapolations, we only make use of the DGP. In the first example, the DGP corresponds to a supplementary additively linked feature x_2:

$$f(x_1, x_2) = \begin{cases} x_1 + x_2 & x_1 < 0 \\ 5\sin(2x_1) + x_2 & x_1 \geq 0 \end{cases}$$

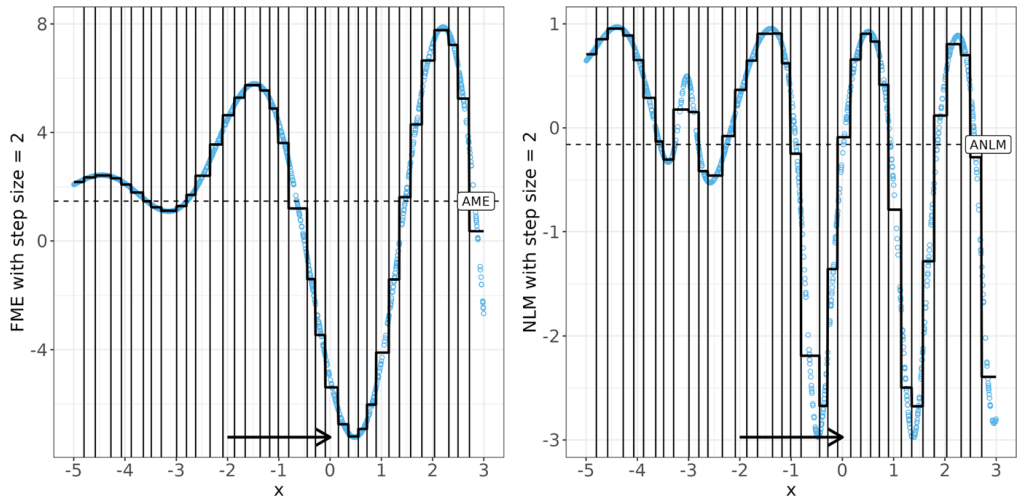


Fig. 9 Univariate data with noise. For each point, moving in x direction by the length of the arrow results in the FME / NLM indicated on the vertical axis. **Left**: FMEs with step size $h = 2$ and cAMEs. **Right**: NLM values and cANLMs (Color figure online)

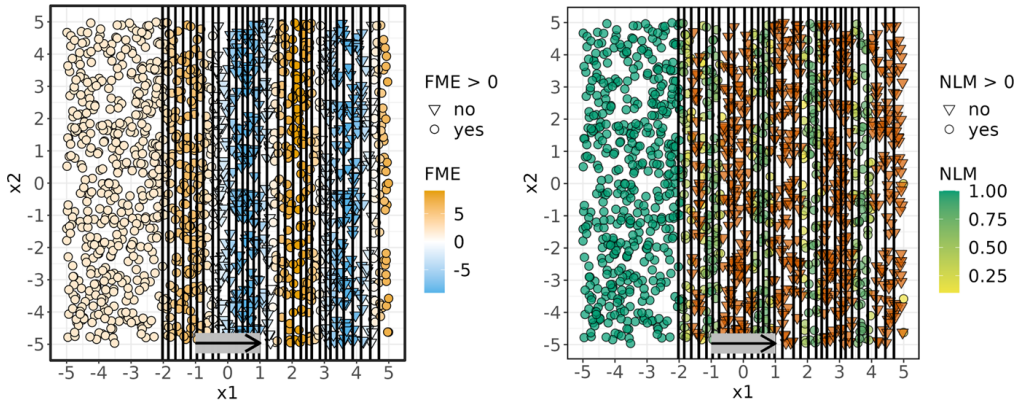
In the second example, the DGP corresponds to a supplementary multiplicatively linked feature x_2, i.e., we have a pure interaction:

$$f(x_1, x_2) = \begin{cases} x_1 \cdot x_2 & x_1 < 0 \\ 5 \sin(2x_1) \cdot x_2 & x_1 \geq 0 \end{cases}$$

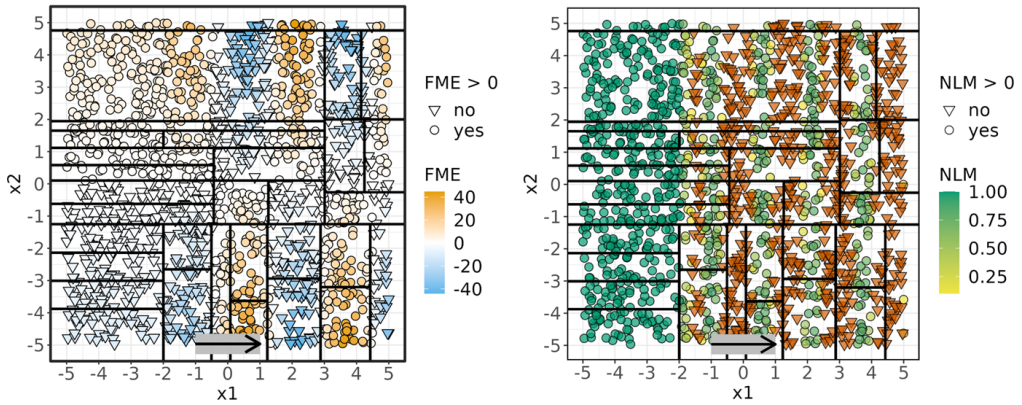
The FMEs and NLM values for both DGPs are given in Fig. 10. For the additive DGP, given the value of x_1, moving in x_2 direction does not influence the FMEs due to the additive recovery property. As a result, we receive the same FMEs with an additively linked feature x_2 as without it (as long as the feature change does not occur in x_2). For the multiplicative DGP, the FMEs now vary for a given x_1 value, even though the feature change only occurs in x_1. The NLM values are both affected by the presence of an additively linked and a multiplicatively linked feature x_2, even though the feature change only occurs in x_1. As opposed to the additive DGP, the cAME tree makes use of x_2 as a split variable for the multiplicative DGP.

6.4 Bivariate data with bivariate feature change

Next, we demonstrate bivariate FMEs and the corresponding NLM. We use the same DGPs as for the univariate feature change. The FMEs and NLM values are given in Fig. 11. As opposed to the univariate feature change for additively linked data, the FME values now also vary in x_2 direction for a given x_1 value due to the simultaneous change in x_2. The NLM indicates linearity for a multitude of observations, given both the additive and the multiplicative DGP. For these observations, we can infer that multiplying *both* step sizes by a value on the interval [0, 1] results in an equally proportionally reduced FME.



(a) DGP with additive link.



(b) DGP with multiplicative link.

Fig. 10 Bivariate data and univariate feature change $\mathbf{h_1} = \mathbf{2}$. For each point, moving in x_1 direction by the length of the arrow results in the FME / NLM indicated by the color. FMEs (left) and NLM (right). Negative NLM values are red-colored (Color figure online)

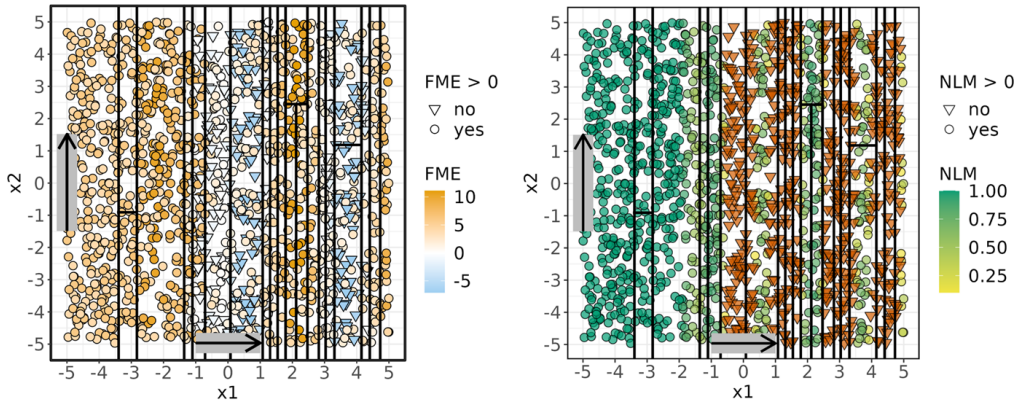
6.5 Friedman's regression problem

In the last simulation example, we demonstrate how FMEs are able to discover effects within a higher-dimensional function.[4] In Friedman's regression problem (Friedman 1991; Breiman 1996), we have 10 independent and uniformly distributed variables on the interval [0, 1]. The target is generated using the first 5 variables:

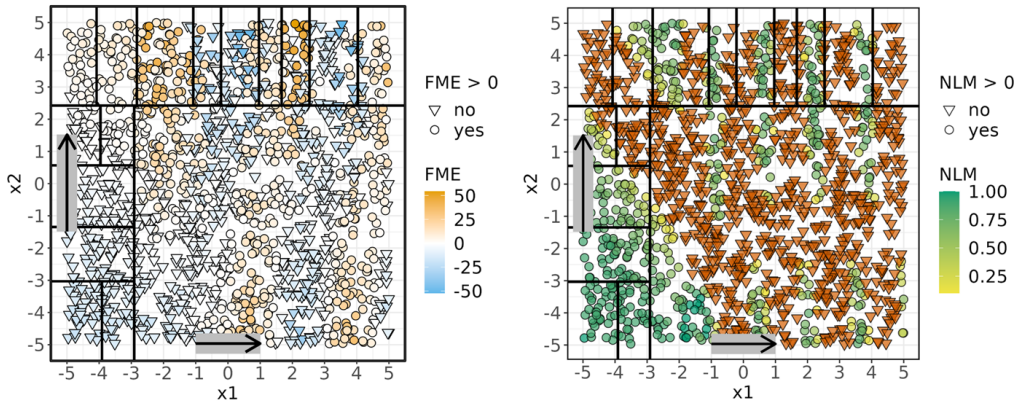
$$y = 10 \sin(\pi x_1 x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + \epsilon$$

where ϵ is drawn from $N(0, \sigma)$. We simulate 1000 instances with $\sigma = 0$ and tune the regularization and sigma parameters of an SVM with a radial basis function kernel on all 10 features. Recall that our ability to conduct inference regarding the DGP depends on how well the model approximates it. In the following illustrations, we select an

[4] As our goal is to recover terms within the DGP, we refrain from computing cAMEs here.



(a) DGP with additive link.



(b) DGP with multiplicative link.

Fig. 11 Bivariate data and bivariate feature change $\mathbf{h_1 = 2}$ and $\mathbf{h_2 = 3}$. For each point, moving in x_1 and x_2 directions by the lengths of the respective arrows results in the FME / NLM indicated by the color. FMEs (left) and NLM (right). Negative NLM values are red-colored (Color figure online)

identical step size of 0.1 for each feature. As this represents roughly 10% of each feature's range, it facilitates the comparison between FMEs and the expected effect within the DGP. For instance, with a step size of 0.1 for x_5, we expect an AME of $5 \cdot 0.1 = 0.5$ if the model has a good fit. In this example, negative NLM values are set to zero (which acts as a hard bound for non-linearity) to compute the ANLM.

We first analyze the interaction pair x_1 and x_2 (see Fig. 12). For small values of either x_1 or x_2, univariate FMEs are mostly positive, while for feature values larger than 0.5, they are increasingly negative. Bivariate FMEs are largest for medium value combinations of x_1 and x_2 or large values of one feature and small values of the other. FMEs are negative for the product of x_1 and x_2 approaching 1. This, too, is expected since the sinus curve's point of inflection is located at $\frac{\pi}{2}$, and the blue area of negative FMEs roughly corresponds to $\frac{\pi}{2} = \pi x_1 x_2$, e.g., for $x_1 = x_2 \approx 0.707$. The bivariate NLM confirms our analysis by indicating strong non-linearity in said area of the sinus curve's point of inflection.

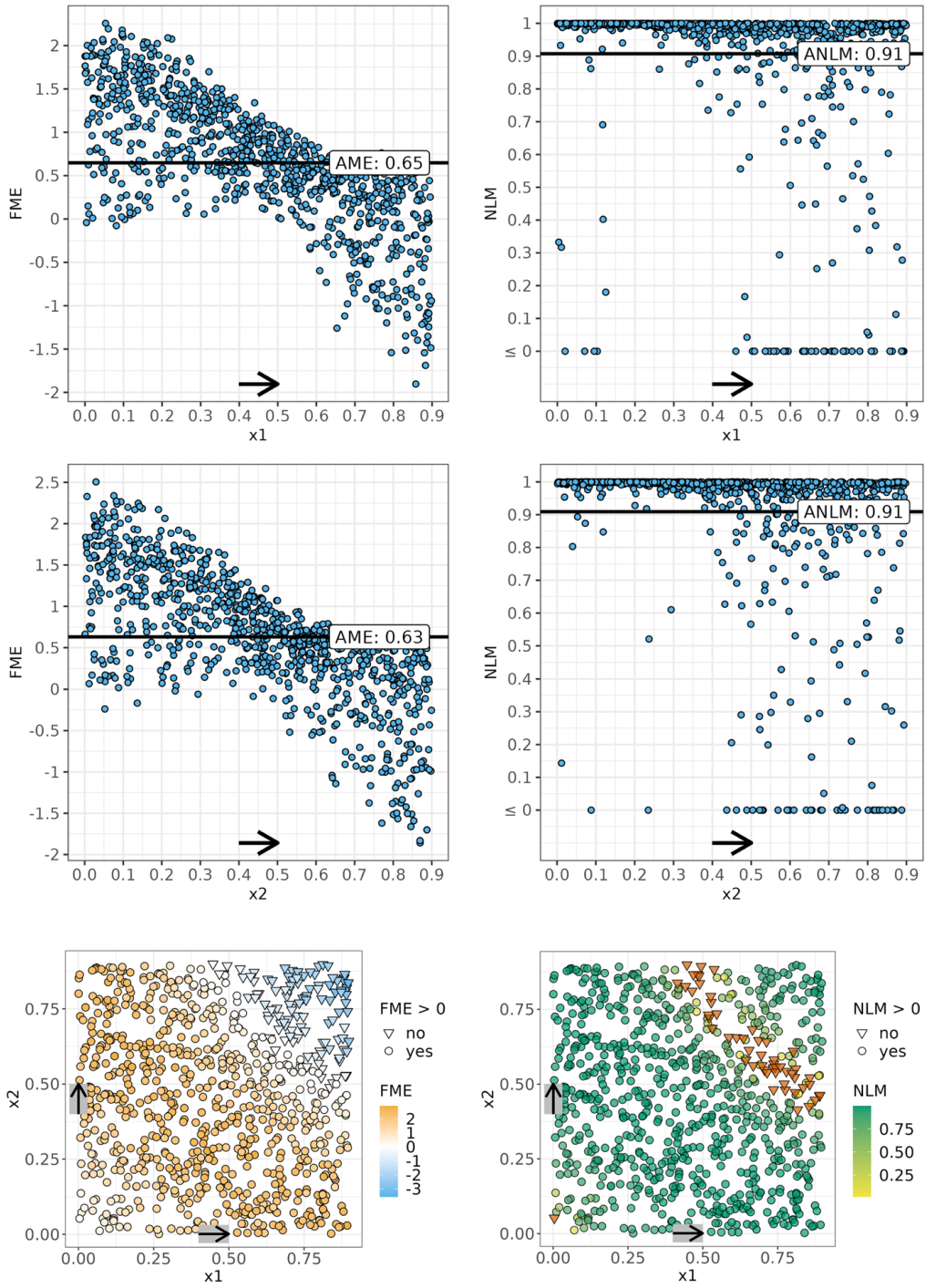


Fig. 12 Friedman's regression problem: Univariate and bivariate FMEs and NLMs for x_1 and x_2 with step sizes of 0.1 for both features. Negative NLM values are red-colored. Around the sinus curve's point of inflection, FMEs turn negative, and the NLM clearly diagnoses non-linearity (red triangles)) (Color figure online)

Next, we evaluate univariate effects of x_3, x_4, and x_5 (see Fig. 13). For x_3, we can see a linear trend of FMEs, which are mostly negative for values smaller than 0.5 and positive for values larger than 0.5. This is expected, since the effect of x_3 within the DGP is quadratic but shifted by 0.5 to the right. The NLM correctly diagnoses strong linearity for small and large values of x_3 but non-linearity for the point of inflection. Both x_4 and x_5 have positive linear effects on the target within the DGP, with the effect of x_4 being twice as large as the effect of x_5. Given the DGP, we would expect an increase of 0.1 in x_4 to have an AME of 1 (observed AME = 0.92) and an increase of 0.1 in x_5 to have an AME of 0.5 (observed AME = 0.46). FMEs reveal both linear patterns with the AMEs closely recovering expected effects and the NLMs indicating strong linearity.
Lastly, we evaluate FMEs for x_6 (see Fig. 14) which has no effect on the target within the DGP. We can see a cluster of FMEs, roughly without any correlations. The AME is approximately zero, thus accurately recovering the (non-existent) feature effect of x_6.

7 Application workflow and applied example

We now present a structured application workflow that incorporates the theory presented in the preceding sections and apply it to real data:

1. Train and tune a predictive model.
2. Based on the application context, choose evaluation points $\mathscr{D}$, the features of interest S, and the step sizes $\boldsymbol{h}_S$.
3. Check whether any $\boldsymbol{x}^{(i)}$ or $(\boldsymbol{x}_S{}^{(i)} + \boldsymbol{h}_S, \boldsymbol{x}_{-S}{}^{(i)})$ are subject to model extrapolations. See Appendix A.2 for possible options.
4. Either modify step sizes so no points are subject to model extrapolations or remove the ones that are.
5. Compute FMEs for selected observations and the chosen step sizes.
6. Optional: Compute the NLM for every computed FME.
7. Optional: Compute cAMEs by finding subgroups with homogeneous FMEs.
8. Optional: Compute cANLM values.
9. Optional: Compute CIs for cAME and cANLM.
10. Conduct local (single FMEs of interest) and (optionally) regional interpretations (cAME and cANLM).

The white wine data set (Cortez et al. 2009) consists of 4898 white wines produced in Portugal. The target is the perceived preference score of wine testers on a scale of 1-10, which we model as a continuous variable. The features consist of wine characteristics such as alcohol by volume (ABV) or the pH value. We start by tuning the regularization and sigma parameters of an SVM with a radial basis function kernel.
We first compare our results to the analysis by Goldstein et al. (2015) who train a neural network with 3 hidden units. They note that their model might be subject to performance issues and that their analysis shall only exemplify the types of interpretations ICE curves are able to generate. Model-agnostic interpretations are conditional on the trained model and can only be vaguely compared. In their analysis, the effect

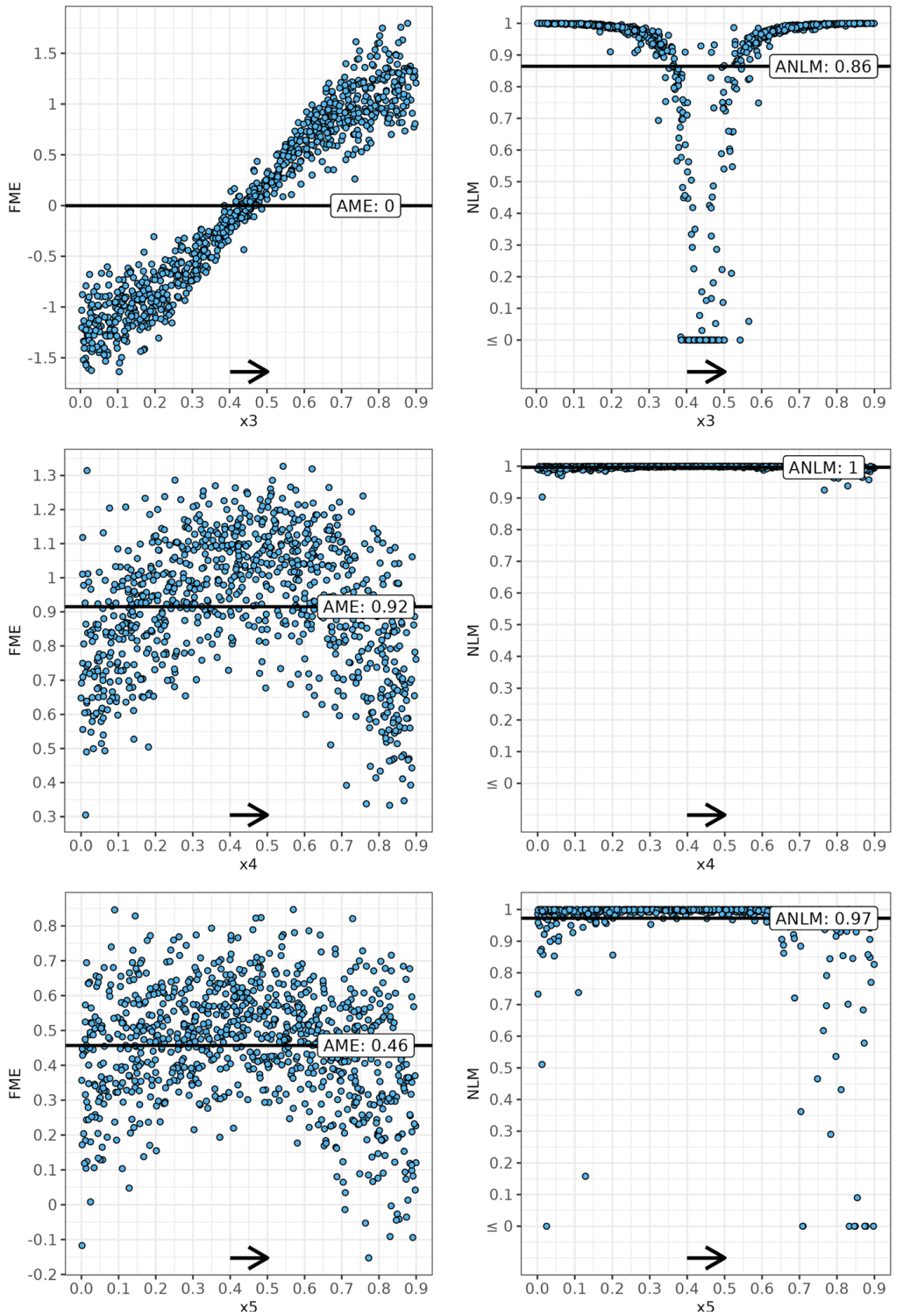


Fig. 13 Friedman's regression problem: Univariate FMEs and NLMs with step sizes of 0.1 for features x_3, x_4, and x_5. The NLM indicates non-linearity around the point of inflection of the quadratic effect of x_3. It indicates strong linearity for x_4 and x_5 which have linear effects on the simulated target. AMEs approximately recover the expected FME within the DGP for x_4 and x_5 (Color figure online)

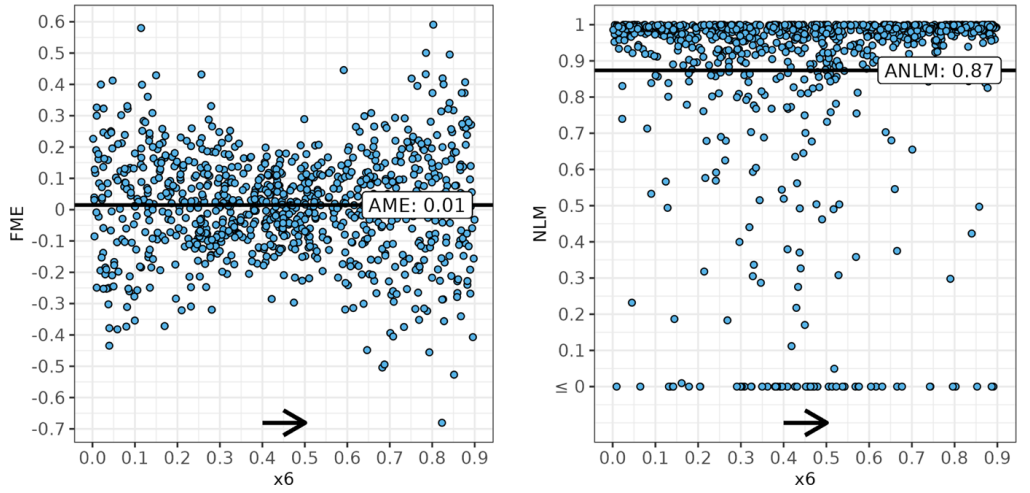


Fig. 14 Friedman's regression problem: Univariate FMEs and NLMs for feature x_6 with a step size of 0.1. FMEs do not exhibit any pattern, and the AME is approximately zero. This correctly diagnoses that x_6 has no effect on the simulated target (Color figure online)

of increasing the pH value on the predicted wine rating differs regarding the wine's alcohol content. In Fig. 15, we compute univariate FMEs of the pH value for a step size of 0.3 (range 2.72 to 3.82). Wines that fall outside the multivariate envelope of the training data are excluded from the analysis. The AME ≈ 0 suggests there is no global feature effect. We use CTREE to search for exactly one split and observe that a wine's alcohol content induces subgroups of a positive cAME of 0.12 (low alcohol) and a negative cAME of -0.39 (high alcohol). Resampling 500 times with 63.2% of the data results in the same split every time. This confirms our proposition that global aggregations are generally not a good descriptor of feature effects and that dividing the data into subgroups lets us discover varying cAMEs. Our methods add new insights compared to ICEs by automatically detecting the interaction between the pH value and alcohol content.

Next, we are interested in the effects of alcohol on a wine's quality rating. Again, the univariate AME of ≈ 0.06 suggests there is a negligible global feature effect. Recall that we motivate FMEs as a local model explanation method first and foremost, which can be extended to regional or global explanations when multiple FMEs are considered. We select a single wine with an ABV of 10.7 (range 8.0 to 14.2) and compute an LLTR for its alcohol content with an NLM threshold value of 0.9. Figure 16 visualizes each explored step size and the corresponding FME and NLM pair. Step sizes that are associated with non-linear effects are greyed out. Indeed, we can observe a large effect on this wine's predicted quality rating given variations in its alcohol content. This confirms our proposition that aggregations of individual FMEs to the AME are not accurately representing feature effects for non-linear models and that evaluating effects for single observations in isolation can provide more insights into the model's workings.

Let us now investigate interactions between both features, first extending our earlier search for an LLTR from Fig. 16 to bivariate step sizes for the same wine, where steps represent 20% of each feature's IQR. We succeed in finding step size combinations

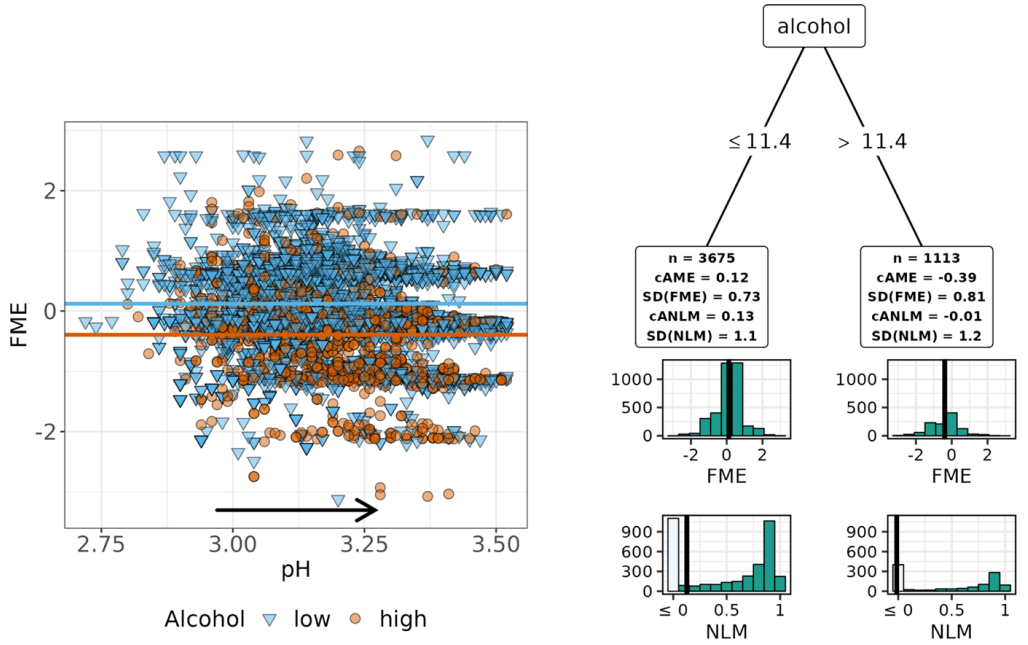


Fig. 15 White wine data: FMEs of increasing a wine's pH value by 0.3 on its perceived quality rating, colored by subgroup found by CTREE (left). The colored horizontal lines indicate cAMEs. CTREE finds subgroups whose cAMEs correspond to 0.12 for ABV $\leq$ 11.4 and -0.39 for ABV $>$ 11.4 (right). A similar interaction was found by Goldstein et al. (2015)

that are associated with linear multivariate effects. Next, we evaluate how the data set behaves as a whole, starting with an exploratory analysis of bivariate step sizes and visualizations of FME and NLM distributions via boxplots (see Fig. 18). For combinations of larger step sizes, we can see a large variance in effects. Analyzing the evolution of boxplots through increasing step sizes, we gather that given low pH values, wine quality ratings are driven by the wine's alcohol content (resulting in an increasing dispersion of FMEs for increases in ABV); given high pH values, increasing the alcohol content has a negligible effect on the wine rating (where the dispersion of FMEs for increases in ABV stays roughly the same). Figure 19 visualizes the bivariate distribution of FMEs over both features given a fixed combination of step sizes (+ 0.3 in pH value and +1% in ABV). The largest effects of such a bivariate increase in feature values are mostly located around lower to medium feature value combinations, whereas FMEs are increasingly negative around higher value combinations.

Lastly, we demonstrate how multivariate FMEs can provide insights into the model's workings when other techniques such as ICEs fail, as they are restricted to univariate and bivariate visualizations. In addition to the previous bivariate feature change, we add a 0.5 $\frac{g}{dm^3}$ increase to the potassium sulphate concentration (range 0.22 to 1.08). This noticeably boosts FMEs. In Fig. 20 we visualize the FME density for the threeway feature change and the corresponding AME and ANLM. Again, the AME would obfuscate interpretations by suggesting a negligible effect of this trivariate feature change on the predicted wine quality rating. In contrast to restrictive techniques such as the ICE and PD, we can take advantage of the FME distilling feature effects into a single value for arbitrary feature changes.

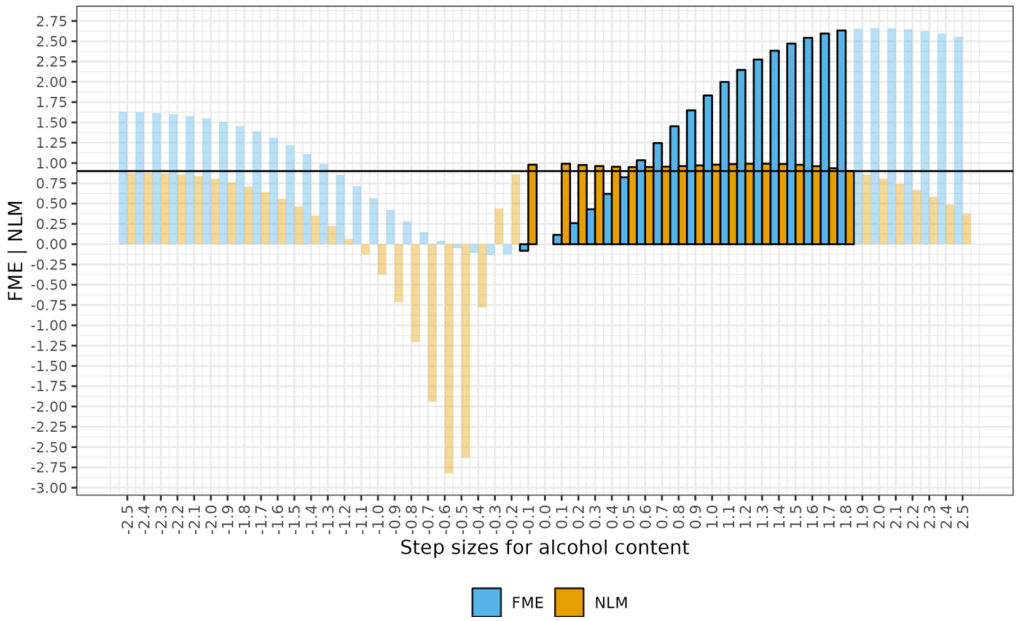


Fig. 16 White wine data: Given a single wine (ABV = 10.7), we compute an LLTR (NLM threshold = 0.9) for changes in ABV. Step sizes that are associated with non-linear effects are greyed out

To sum up, we discover that the pH value influences predicted wine quality ratings on a global scale and that the effect differs depending on a wine's alcohol content. ABV has large local effects on predicted wine quality ratings, which cancel each other out when being averaged to an AME. For single observations, we can find trust regions for linear effects. There is an interaction between the pH value and alcohol content with intensely varying effects across observations. The LLTR for ABV can be extended to bivariate changes in pH value and alcohol content for the same, single wine. Furthermore, there are large multimodal effects when adding a third feature change in the potassium sulphate concentration where—again—the AME obfuscates interpretations by indicating a negligible global feature effect.

8 Conclusion

This research paper introduces FMEs as a model-agnostic interpretation method for arbitrary prediction functions, e.g., in the context of ML applications. We create a unified definition of FMEs for both univariate and multivariate, as well as continuous, categorical, and mixed-type features. Furthermore, we introduce an NLM for FMEs based on the similarity between the prediction function and the intersecting linear secant. Due to the complexity and non-linearity of ML models, we suggest to focus on regional instead of global feature effects. We propose a means of estimating expected conditional FMEs via cAMEs and present one strategy to find population subgroups by partitioning the feature space with decision trees. The resulting subgroups can be augmented with cANLM values and CIs in order to receive a compact summary of

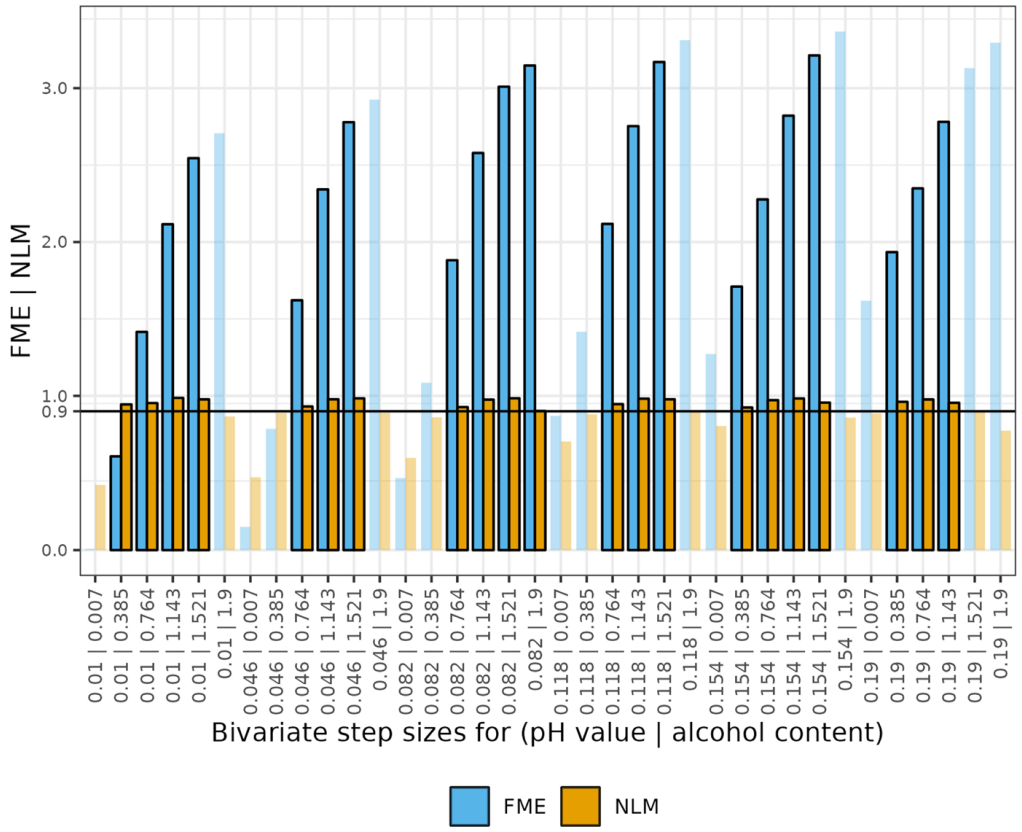


Fig. 17 White wine data: We select a single observation (i.e., a single wine) and compute an LLTR for bivariate step size combinations of the pH value and ABV with an NLM threshold of 0.9. Step size combinations that are associated with non-linear effects are greyed out

the prediction function across the feature space. In the Appendix, we provide proofs on the additive recovery property of FMEs and their relation to the ICE and PD.

Given arbitrary predictive models, FMEs can be used to address questions on a model's behavior such as the following: Given pre-specified changes in one or multiple feature values, what is the expected change in predicted outcome? What is the change in prediction for an average observation? What is the change in prediction for a pre-specified observation? What are population subgroups with more homogeneous average effects? What is the degree of non-linearity in these effects? What is our confidence in these estimates? What is the expected change in prediction when switching observed categorical feature values to a reference category?

However, model-agnostic interpretation methods are subject to certain limitations. They are favorable tools to explain the model behavior but often fail to explain the underlying DGP, as the quality of the explanations relies on the closeness between model and reality. Molnar et al. (2022) discuss various general pitfalls of model-agnostic interpretation methods, e.g., model extrapolations, estimation uncertainty, or unjustified causal interpretations.

Throughout the manuscript, we noted various directions that may be explored in future work. For the selection of step sizes, one may work towards better quantifying extrap-

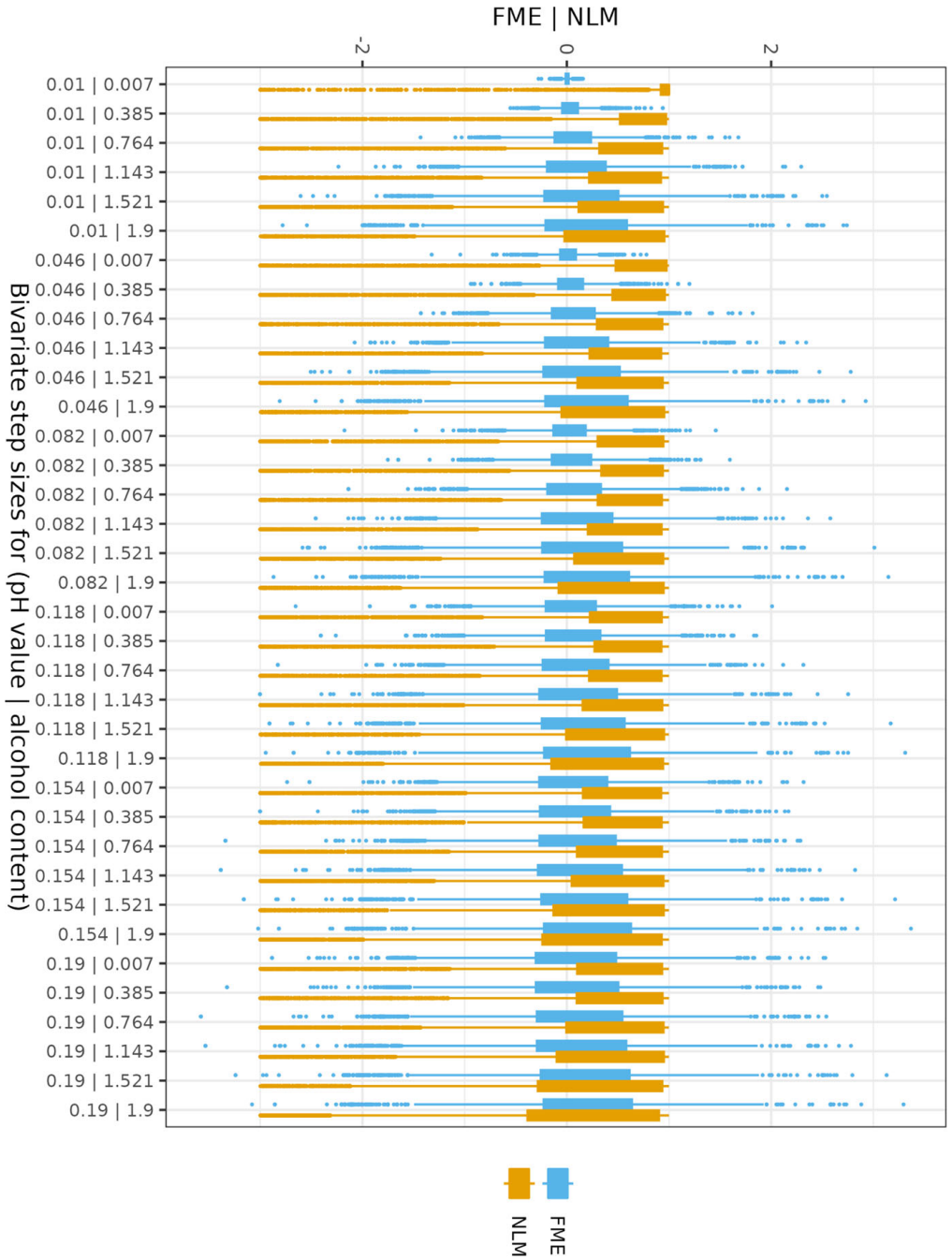


Fig. 18 White wine data: Here, we explore how bivariate step sizes affect the global distribution of FME / NLM values. Such an analysis may provide hints about what step sizes or step size combinations drive effects in the model. With visualizations such as in Fig. 19, we can then "zoom in" on a particular step size combination

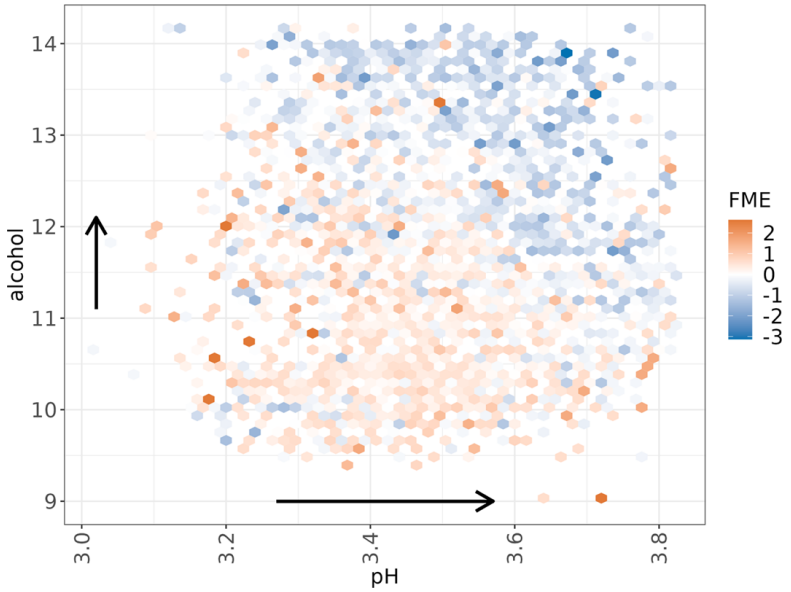


Fig. 19 White wine data: Distribution of FMEs given pH and alcohol values. We use averages within hexagons to avoid overplotting values. FME hexagon averages are mostly positive around lower to medium value combinations of both features, while they are increasingly negative around higher value combinations

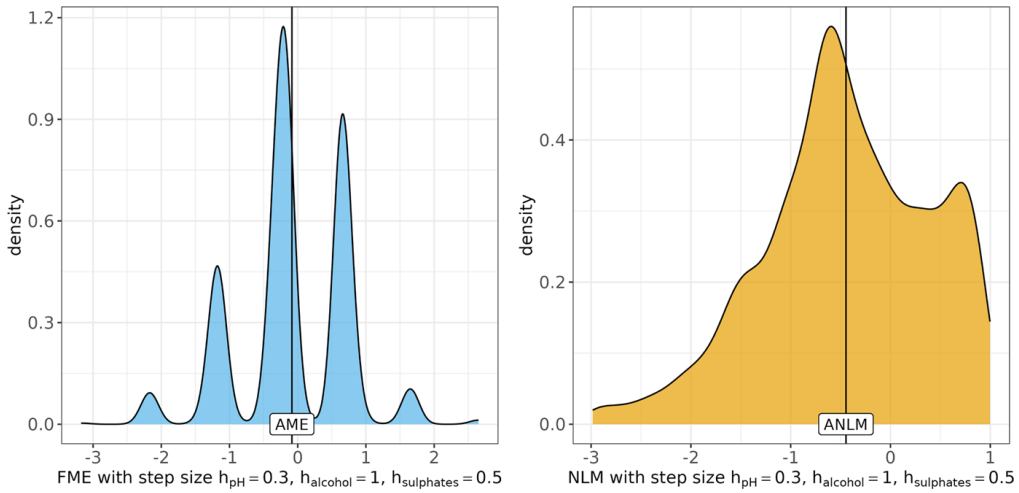


Fig. 20 White wine data: Demonstrating how FMEs can be used to interpret the model for threeway interactions when other techniques such as ICEs fail. We evaluate distributions of FMEs for feature changes of the pH value by 0.3, ABV by 1%, and the potassium sulphate concentration by 0.5 $\frac{g}{dm^3}$. FMEs are multimodal. Plotting the corresponding NLM distribution reveals considerable non-linearity for the majority of trivariate FMEs

olation risk. For subgroup selection, one may work towards stabilizing split search or quantifying subgroup uncertainty. To spare computations or facilitate local interpretations, one may search for a subset of representative observations. Furthermore, FMEs may be used for feature importance computations as well.
Many disciplines that have been relying on traditional statistical models—and interpretations in terms of MEs, the AME, MEM, or MER—are starting to utilize the predictive power of ML. With this research paper, we aim to bridge the gap between the restrictive theory on MEs with traditional statistical models and the more flexible and capable approach of interpreting modern ML models with FMEs.

A Background information

A.1 Decomposition of the prediction function

The prediction function to be analyzed may be very complex or even a black box. However, there are multiple ways to decompose the prediction function into a sum of components of increasing order. Although the goal of FMEs is not to decompose the prediction function, it is convenient to either regard the prediction function as an additive decomposition or to keep in mind that it may be decomposed into one. An additive decomposition of the prediction function has the following general form:

$$\widehat{f}(\boldsymbol{x}) = g_{\{0\}} + g_{\{1\}}(x_1) + g_{\{2\}}(x_2) + \cdots + g_{\{1,2\}}(x_1, x_2) + \cdots + g_{\{1,\ldots,p\}}(\boldsymbol{x}) \quad (3)$$

In SA, the additive decomposition is typically referred to as a high-dimensional model representation (HDMR) or ANOVA-HDMR (Saltelli et al. 2008). Various approaches exist to estimate Eq. (3) or a truncated variant, e.g., via recursive computations of PD functions (Hooker 2004b, 2007), random sampling HDMR (Li et al. 2006), or accumulated local effects (Apley and Zhu 2020). Further assumptions are needed to make the decomposition unique, e.g., feature independence (Chastaing et al. 2012). For instance, we may recursively compute Eq. (3) as follows:

$$\begin{aligned}
g_{\{0\}} &= \mathbb{E}_{\boldsymbol{X}}\left[\widehat{f}(\boldsymbol{X})\right] \\
g_{\{1\}}(x_1) &= \mathbb{E}_{\boldsymbol{X}_{-1}}\left[\widehat{f}(x_1, \boldsymbol{X}_{-1})\right] - g_{\{0\}} \\
g_{\{2\}}(x_2) &= \mathbb{E}_{\boldsymbol{X}_{-2}}\left[\widehat{f}(x_2, \boldsymbol{X}_{-2})\right] - g_{\{0\}} \\
g_{\{1,2\}}(x_1, x_2) &= \mathbb{E}_{\boldsymbol{X}_{-\{1,2\}}}\left[\widehat{f}\left(x_1, x_2, \boldsymbol{X}_{-\{1,2\}}\right)\right] - g_{\{2\}}(x_2) - g_{\{1\}}(x_1) - g_{\{0\}} \\
&\vdots \\
g_{\{1,\ldots,p\}}(\boldsymbol{x}) &= \widehat{f}(\boldsymbol{x}) - \cdots - g_{\{1,2\}}(x_1, x_2) - g_{\{2\}}(x_2) - g_{\{1\}}(x_1) - g_{\{0\}}
\end{aligned}$$

where $\mathbb{E}_{\boldsymbol{X}_{-S}}\left[\widehat{f}(\boldsymbol{x}_S, \boldsymbol{X}_{-S})\right]$ is typically referred to as the PD of $\widehat{f}$ on feature set S in ML. Model decompositions are frequently used in variance-based SA. We refer the reader to the overview by Saltelli et al. (2008) for more details.

A.2 Model extrapolation

King and Zeng (2006) define extrapolation as predicting outside the convex hull of the training data. They demonstrate that the task of determining whether a point is located inside the convex hull can be efficiently solved using linear programming. However, the convex hull may be comprised of many empty areas without training observations, especially in the case of correlated and high-dimensional data. Therefore, it seems plausible to define model extrapolation differently, e.g., as predictions in areas of the feature space with a low density of training points. Hooker (2004a) summarizes two main predicaments of model extrapolations. First, the model creates predictions which do not accurately reflect the target distribution given the features. Second, the predictions are subject to a high variance. Many model-agnostic techniques are subject to model extrapolation risks (Molnar et al. 2022). Hooker (2007) warns against model extrapolations when computing model decompositions. Hooker et al. (2021) call attention to the perils of permuting feature values for feature importance computations. It is important to note that this issue highly depends on the behavior of the chosen model. The issue of determining whether the model extrapolates essentially boils down to quantifying the prediction uncertainty. Some models might diverge considerably from a scenario where they would have been supplied with enough training data (high prediction uncertainty), while other models might be relatively robust against such issues (low prediction uncertainty). Although FMEs based on model extrapolations are still correct in terms of the model output, they might not represent any underlying DGP in an accurate way. Therefore, it is important to take into account (and preferably avoid) potential model extrapolations when selecting feature values and step sizes to compute FMEs.

For some models, built-in measures exist to quantify the prediction uncertainty (Munson and Kegelmeyer 2013), e.g., the proximity measure for tree ensembles which counts how often a pair of points is located in the same leaf node for all trees of the ensemble (Hastie et al. 2001). The same can be done for the pairwise proximity between points in the training and the test set. For instance, given n training observations and a test observation $\boldsymbol{x}$, we can create an $(n \times 1)$ vector of proximities which can be used to detect model extrapolations. However, it is desirable to detect model extrapolations via auxiliary extrapolation risk metrics (AERM) (Munson and Kegelmeyer 2013) which are independent of the trained model. Detecting an EP is similar in concept to the detection of outliers. Although a unified definition of outliers does not exist, they are generally considered to differ as much from other observations as to suspect they were generated by a different mechanism (Hawkins 1980). We can therefore consider an outlier to be drawn from a different distribution than the training data (and one that does not overlap with it), which suits our definition of EPs. In clustering, outliers are often found using local density-based outlier scores such as local outlier probabilities (LOP) (Kriegel et al. 2009). Based on the nearest data points, LOP provides an interpretable score on the scale [0, 1], indicating the probability of a point being an outlier. However, clustering techniques such as LOP are often based on the assumption that the data exhibits a structure of clusters or on assumptions about the clusters' distributions. In theory, one could use various other outlier detection (also referred to

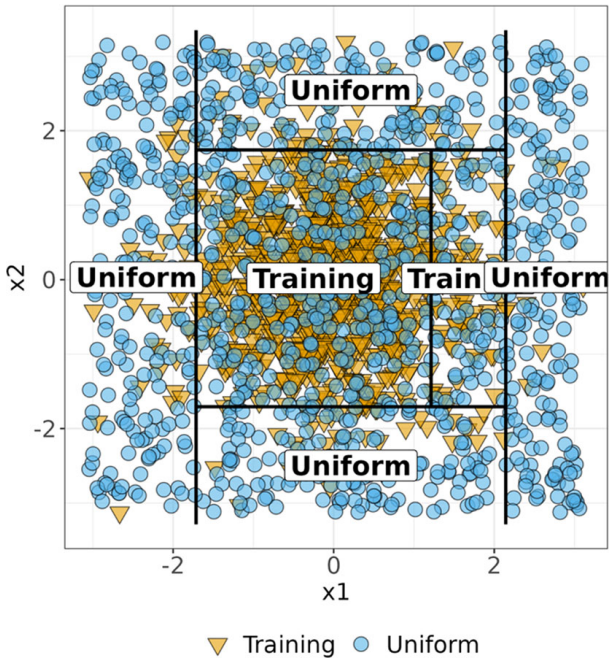


Fig. 21 We augment the training data (orange) with uniform points (blue). A classification tree partitions the feature space into non-extrapolation areas (predominantly occupied with training observations) and extrapolation areas (predominantly occupied with uniform Monte-Carlo samples)

as anomaly detection) mechanisms for extrapolation detection, e.g., isolation forests (Liu et al. 2012).

Hooker (2004a) proposes a statistical test to classify a point as an EP or non-EP. It tests whether a point was more likely to be drawn from the data distribution (non-EP) or the uniform distribution (EP). The uniform distribution is used as an uninformative baseline distribution. The extrapolation risk indicator $R(\boldsymbol{x})$ corresponds to:

$$R(\boldsymbol{x}) = \frac{U(\boldsymbol{x})}{U(\boldsymbol{x}) + P(\boldsymbol{x})} \tag{4}$$

with $U(\boldsymbol{x})$ being the density function of the uniform distribution and $P(\boldsymbol{x})$ the density function of the data distribution. $R(\boldsymbol{x})$ has a range of $[0, 1]$ with 0 indicating the lowest and 1 the highest extrapolation risk. $R(\boldsymbol{x}) > 0.5$ indicates extrapolation. As the support of $U(\boldsymbol{x})$ we may either choose the recommendations of an application domain expert or the observed feature ranges. Equation (4) cannot be directly computed, as the density of the training data is unknown. If $\boldsymbol{x}$ falls outside the multivariate envelope of the training data, it is plausible to set $R(\boldsymbol{x})$ to 1.

We may estimate Eq. (4) by creating a binary classification problem on a data set augmented with uniform Monte-Carlo samples (Hooker 2004a). The training data is labeled as the foreground class. Next, artificial data points are sampled from a uniform distribution and labeled as the background class. A predictive model is trained on the augmented data set and predicts for a given point whether it is more probable that it was drawn from the data distribution or the uniform distribution. Consider two independent standard normally distributed features. We augment the training data with a uniform Monte-Carlo sample with support $[min(x_1), max(x_1)] \times [min(x_2), max(x_2)]$ and use CART to partition the feature space into extrapolation areas and non-extrapolation

areas (see Fig. 21). Some training points are located outside the center rectangles in a low-density end of the bivariate normal distribution. Therefore, it is correct to be cautious when evaluating predictions in this area, even if a point was drawn from the training data.
Hooker (2004a) argues that in high-dimensional settings, the Monte-Carlo sample will leave lots of areas of the feature space unoccupied which results in poor classification performance. Classification performance may be boosted by directly utilizing distributional information about the uniform distribution instead of a Monte-Carlo sample. This technique termed confidence and extrapolation representation trees (CERT) exploits a property of classification trees which lets one replace the number of Monte-Carlo points per subspace with the expected number of uniform points at each split. Given the feature space $\mathscr{X}$ with n observations and a subspace $\mathscr{X}_{[\,]}$ with $n_{[\,],\text{data}}$ observations, the expected number of uniform points on the subspace $n_{[\,],\,\text{uniform}}$ is proportional to the fraction of feature space hypervolume the subspace occupies:

$$n_{[\,],\,\text{uniform}} = \frac{\text{hypervolume}(\mathscr{X}_{[\,]})}{\text{hypervolume}(\mathscr{X})} \cdot n_{[\,],\,\text{data}}$$

For the tree growing and pruning strategy, CERT uses a mixture of both CART (e.g., splitting based on the Gini index) and C4.5 (Quinlan 1993) (e.g., missing values and surrogate splits). Apart from letting us directly supply the classification tree with distributional information instead of data, its interpretability is advantageous. The tree partitions the entire feature space at once into hyperrectangles that indicate extrapolation or non-extrapolation areas. Hooker (2004a) argues that CERT provides a markedly lower misclassification rate as opposed to using Monte-Carlo samples with a classification tree. However, it is unclear whether this advantage holds for other classification algorithms used with Monte-Carlo samples.

A.3 Marginal effects for tree-based prediction functions

DMEs are not suited to interpret piecewise constant prediction functions, e.g., classification and regression trees (CART) or tree ensembles such as random forests or gradient boosted trees. Generally, most observations are located on piecewise constant parts of the prediction function where the derivative equals zero. FMEs provide two advantages when interpreting tree-based prediction functions: First, a large enough step size will often involve traversing a jump discontinuity (which corresponds to a tree split in RP) on the prediction function (see Fig. 22), so the FME does not equal zero; second, measures of spread such as the variance can indicate what fraction of FMEs traversed a jump discontinuity and what fraction did not.

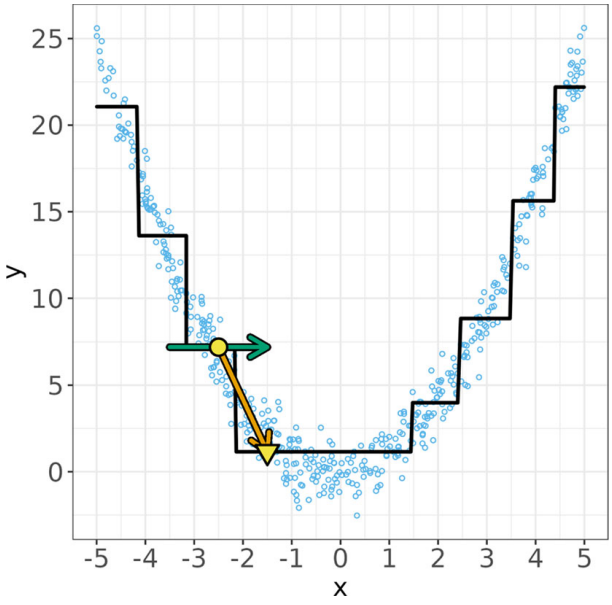


Fig. 22 A quadratic relationship between the target y and a single feature x. A decision tree fits a piecewise constant prediction function (black line) to the training data (blue points). The DME (slope of green arrow) at the point $x = -2.5$ (yellow dot) is zero, while the FME with $h = 1$ traverses the jump discontinuity (secant = orange arrow) and reaches the point $x = -1.5$ (yellow triangle)

B Proofs

B.1 Additive recovery

We provide several proofs on additive recovery based on a prediction function in additive form. Any prediction function can be decomposed into a sum of effect terms of various orders (see Appendix 1). The sum of effect terms of a feature set K is denoted by $\Theta_K(\boldsymbol{x}_K)$. For notational simplicity, the union $\{j\} \cup K$ of the j-th feature index and the index set K is denoted by $\{j, K\}$. The sum of effect terms is denoted by $\Theta_{\{j,K\}}(x_j, \boldsymbol{x}_K)$.

Theorem 1 *(Additive Recovery of Finite Difference) An FD w.r.t. x_j only recovers terms that depend on x_j and no terms that exclusively depend on $\boldsymbol{x}_{-j}$.*

Proof Consider a prediction function $\widehat{f}$ that consists of a sum, including the main effect of x_j, denoted by $g_{\{j\}}(x_j)$, a sum of higher order terms (interactions) between x_j and other features $\boldsymbol{x}_K$, denoted by $\Theta_{\{j,K\}}(x_j, \boldsymbol{x}_K)$, and terms that depend on the remaining features $x_{-\{j,K\}}$, denoted by $\Theta_{-\{j,K\}}(\boldsymbol{x}_{-\{j,K\}})$:

$$\widehat{f}(\boldsymbol{x}) = g_{\{j\}}(x_j) + \Theta_{\{j,K\}}(x_j, \boldsymbol{x}_K) + \Theta_{-\{j,K\}}(\boldsymbol{x}_{-\{j,K\}})$$

It follows that the FD of predictions corresponds to a function that only depends on x_j, i.e., it locally recovers the relevant terms on the interval $[x_j + a, x_j + b]$.

$$\begin{aligned} FD_{j,\boldsymbol{x},a,b} &= \widehat{f}(x_1, \ldots, x_j + a, \ldots, x_p) - \widehat{f}(x_1, \ldots, x_j + b, \ldots, x_p) \\ &= \left[g_{\{j\}}(x_j + a) \; + \; \Theta_{\{j,K\}}(x_j + a, \boldsymbol{x}_K) \; + \; \Theta_{-\{j,K\}}(\boldsymbol{x}_{-\{j,K\}})\right] \\ &\quad - \left[g_{\{j\}}(x_j + b) \; + \; \Theta_{\{j,K\}}(x_j + b, \boldsymbol{x}_K) \; + \; \Theta_{-\{j,K\}}(\boldsymbol{x}_{-\{j,K\}})\right] \\ &= g_{\{j\}}(x_j + a) - g_{\{j\}}(x_j + b) + \Theta_{\{j,K\}}(x_j + a, \boldsymbol{x}_K) \\ &\quad - \Theta_{\{j,K\}}(x_j + b, \boldsymbol{x}_K) \end{aligned}$$

□

Corollary 1 *(Additive Recovery of Univariate Forward Marginal Effect) The univariate FME w.r.t. x_j only recovers terms that depend on x_j and no terms that exclusively depend on $\boldsymbol{x}_{-j}$.*

Proof Consider a prediction function $\widehat{f}$ that consists of a sum, including the main effect of x_j, denoted by $g_{\{j\}}(x_j)$, a sum of higher order terms (interactions) between x_j and other features $\boldsymbol{x}_K$, denoted by $\Theta_{\{j,K\}}(x_j, \boldsymbol{x}_K)$, and terms that depend on the remaining features $\boldsymbol{x}_{-\{j,K\}}$, denoted by $\Theta_{-\{j,K\}}(\boldsymbol{x}_{-\{j,K\}})$:

$$\widehat{f}(\boldsymbol{x}) = g_{\{j\}}\left(x_j\right) + \Theta_{\{j,K\}}\left(x_j, \boldsymbol{x}_K\right) + \Theta_{-\{j,K\}}\left(\boldsymbol{x}_{-\{j,K\}}\right)$$

The FD w.r.t. x_j is equivalent to the FME w.r.t. x_j with $a = h_j$ and $b = 0$. Using Theorem 1, it follows that:

$$\text{FME}_{\boldsymbol{x},h_j} = g_{\{j\}}\left(x_j + h_j\right) - g_{\{j\}}\left(x_j\right) + \Theta_{\{j,K\}}\left(x_j + h_j, \boldsymbol{x}_K\right) - \Theta_{\{j,K\}}\left(x_j, \boldsymbol{x}_K\right)$$

□

Theorem 2 *(Additive Recovery of Multivariate Forward Marginal Effect) The multivariate FME w.r.t. $\boldsymbol{x}_S$ only recovers terms that depend on $\boldsymbol{x}_S$ and no terms that exclusively depend on $\boldsymbol{x}_{-S}$.*

Proof Consider a feature set S. The power set of S excluding the empty set is denoted by $\mathscr{P}^* = \mathscr{P}(S) \setminus \emptyset$. The prediction function $\widehat{f}$ consists of a sum, including the sum of effects of all subsets of features $K \in \mathscr{P}^*$, denoted by $\sum_{K \in \mathscr{P}^*} g_K\left(\boldsymbol{x}_K\right)$, and a sum

of terms that depend on the remaining features, denoted by $\Theta_{-S}(\boldsymbol{x}_{-S})$:

$$\begin{aligned}
\widehat{f}(\boldsymbol{x}) &= \sum_{K \in \mathscr{P}^*} g_K\left(\boldsymbol{x}_K\right) + \Theta_{-S}\left(\boldsymbol{x}_{-S}\right) \\
\text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} &= \left[\sum_{K \in \mathscr{P}^*} g_K(\boldsymbol{x}_K + \boldsymbol{h}_K) + \Theta_{-S}(\boldsymbol{x}_{-S})\right] \\
&\quad - \left[\sum_{K \in \mathscr{P}^*} g_K(\boldsymbol{x}_K) + \Theta_{-S}\left(\boldsymbol{x}_{-S}\right)\right] \\
&= \sum_{K \in \mathscr{P}^*} \left[g_K(\boldsymbol{x}_K + \boldsymbol{h}_K) - g_K(\boldsymbol{x}_K)\right]
\end{aligned}$$

□

B.2 Relation between forward marginal effects, the individual conditional expectation, and partial dependence

Theorem 3 *(Equivalence between Forward Marginal Effect and Forward Difference of Individual Conditional Expectation) The FME with step size $\boldsymbol{h}_S$ is equivalent to the forward difference with step size $\boldsymbol{h}_S$ between two locations on the ICE.*

Proof

$$\begin{aligned}
\text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} &= \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S}) - \widehat{f}(\boldsymbol{x}) \\
&= \text{ICE}_{\boldsymbol{x},S}(\boldsymbol{x}_S + \boldsymbol{h}_S) - \text{ICE}_{\boldsymbol{x},S}(\boldsymbol{x}_S)
\end{aligned}$$

□

Theorem 4 *(Equivalence between Average Marginal Effect and Forward Difference of Partial Dependence for Linear Prediction Functions) The AME with step size $\boldsymbol{h}_S$ is equivalent to the forward difference with step size $\boldsymbol{h}_S$ between two locations on the PD for prediction functions that are linear in $\boldsymbol{x}_S$.*

Proof If $\widehat{f}$ is linear in $\boldsymbol{x}_S$:

$$\begin{aligned}
&\widehat{f}\left(\boldsymbol{x}_S{}^{(i)} + \boldsymbol{h}_S\right) = \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S) \quad \forall\ i \in \{1, \ldots, n\}, \\
&\boldsymbol{x}_S,\ \boldsymbol{h}_S \in \times_{j \in S} \mathscr{X}_j
\end{aligned} \tag{5}$$

It follows:

$$\begin{aligned}
\text{AME}_{\mathscr{D},\mathbf{h}_S} &= \frac{1}{n}\sum_{i=1}^{n}\left[\widehat{f}\left({\mathbf{x}_S}^{(i)} + \mathbf{h}_S, {\mathbf{x}_{-S}}^{(i)}\right) - \widehat{f}\left(\mathbf{x}^{(i)}\right)\right] \\
&= \frac{1}{n}\sum_{i=1}^{n}\widehat{f}\left({\mathbf{x}_S}^{(i)} + \mathbf{h}_S, {\mathbf{x}_{-S}}^{(i)}\right) - \frac{1}{n}\sum_{i=1}^{n}\widehat{f}\left({\mathbf{x}_S}^{(i)}, {\mathbf{x}_{-S}}^{(i)}\right) \\
&\overset{(5)}{=} \frac{1}{n}\sum_{i=1}^{n}\widehat{f}\left(\mathbf{x}_S + \mathbf{h}_S, {\mathbf{x}_{-S}}^{(i)}\right) - \frac{1}{n}\sum_{i=1}^{n}\widehat{f}\left(\mathbf{x}_S, {\mathbf{x}_{-S}}^{(i)}\right) \\
&= \widehat{\text{PD}}_{\mathscr{D},S}\left(\mathbf{x}_S + \mathbf{h}_S\right) - \widehat{\text{PD}}_{\mathscr{D},S}\left(\mathbf{x}_S\right)
\end{aligned}$$

□

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability All data are created or provided in the following public repository: https://github.com/scholbeck/forward_marginal_effects.git

Code availability We provide reproducible scripts for our simulations and the applied example in the following public repository: https://github.com/scholbeck/forward_marginal_effects.git

Declarations

Conflict of interest Not applicable.

Ethics approval Not applicable.

Consent to participate Not applicable.

References

Ai C, Norton EC (2003) Interaction terms in logit and probit models. Economics Letters 80(1):123–129

Alt H, Godau M (1995) Computing the Fréchet distance between two polygonal curves. International Journal of Computational Geometry & Applications 05(01n02):75–91

Ancona M, Ceolini E, Öztireli C, Gross M (2018) Towards better understanding of gradient-based attribution methods for deep neural networks. In: International Conference on Learning Representations, https://openreview.net/forum?id=Sy21R9JAW

Apley DW, 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

Arel-Bundock V (2023) marginaleffects: Predictions, Comparisons, Slopes, Marginal Means, and Hypothesis Tests. https://marginaleffects.com/, R package version 0.15.1.9002
Athey S (2017) Beyond prediction: Using big data for policy problems. Science 355(6324):483–485
Bartus T (2005) Estimation of marginal effects using margeff. The Stata Journal 5(3):309–329
Belogay E, Cabrelli C, Molter U, Shonkwiler R (1997) Calculating the Hausdorff distance between curves. Information Processing Letters 64(1):17–22
Bertsimas D, Dunn J (2017) Optimal classification trees. Machine Learning 106(7):1039–1082
Breiman L (1996) Bagging predictors. Machine Learning 24(2):123–140
Breiman L (2001) Random forests. Machine Learning 45(1):5–32
Breiman L (2001b) Statistical modeling: The two cultures. Statist Sci 16(3):199–231, with comments and a rejoinder by the author
Breiman L, Friedman JH, Olshen RA, Stone CJ (1984) Classification and Regression Trees. Wadsworth and Brooks, Monterey, CA
Casalicchio G, Molnar C, Bischl B (2019) Visualizing the feature importance for black box models. In: Berlingerio M, Bonchi F, Gärtner T, Hurley N, Ifrim G (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2018. Lecture Notes in Computer Science, Springer, Cham, vol 11051
Chastaing G, Gamboa F, Prieur C (2012) Generalized Hoeffding-Sobol decomposition for dependent variables - application to sensitivity analysis. Electronic Journal of Statistics 6:2420–2448
Cortez P, Cerdeira A, Almeida F, Matos T, Reis J (2009) Wine Quality. UCI Machine Learning Repository, https://doi.org/10.24432/C56S3T
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. Journal of Machine Learning Research 20(177):1–81
Friedman JH (1991) Multivariate Adaptive Regression Splines. The Annals of Statistics 19(1):1–67
Friedman JH (2001) Greedy function approximation: A gradient boosting machine. The Annals of Statistics 29(5):1189–1232
Gelman A, Pardoe I (2007) Average predictive comparisons for models with nonlinearity, interactions, and variance components. Sociological Methodology 37(1):23–51
Goldstein A, Kapelner A, Bleich J, 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
Greene W (2012) Econometric Analysis. Pearson International Edition, Pearson Education Limited
Hastie T, Tibshirani R, Friedman J (2001) The Elements of Statistical Learning. Springer Series in Statistics, Springer New York Inc
Hawkins DM (1980) Identification of Outliers. Springer, Netherlands,. https://doi.org/10.1007/978-94-015-3994-4_1
Hooker G (2004a) Diagnosing extrapolation: Tree-based density estimation. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery, New York, NY, USA, KDD '04, p 569-574
Hooker G (2004b) Discovering additive structure in black box functions. In: Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, KDD '04, pp 575–580
Hooker G (2007) Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables. Journal of Computational and Graphical Statistics 16(3):709–732
Hooker G, Mentch L, 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):82
Hothorn T, Hornik K, Zeileis A (2006) Unbiased recursive partitioning: A conditional inference framework. Journal of Computational and Graphical Statistics 15(3):651–674
King G, Zeng L (2006) The dangers of extreme counterfactuals. Political Analysis 14(2):131–159
Kriegel HP, Kröger P, Schubert E, Zimek A (2009) LoOP: Local outlier probabilities. In: Proceedings of the 18th ACM Conference on Information and Knowledge Management, Association for Computing Machinery, New York, NY, USA, CIKM '09, p 1649-1652
Last M, Maimon O, Minkov E (2002) Improving stability of decision trees. International Journal of Pattern Recognition and Artificial Intelligence 16(02):145–159

Leeper TJ (2018) margins: Marginal effects for model objects. https://CRAN.R-project.org/package=margins, R package version 0.3.23
Li G, Hu J, Wang SW, Georgopoulos PG, Schoendorf J, Rabitz H (2006) Random sampling-high dimensional model representation (RS-HDMR) and orthogonality of its different order component functions. The Journal of Physical Chemistry A 110(7):2474–2485
Liu FT, Ting KM, Zhou ZH (2012) Isolation-based anomaly detection. ACM Trans Knowl Discov Data 6(1)
Loh WY (2014) Fifty years of classification and regression trees. International Statistical Review 82(3):329–348
Lundberg SM, Lee SI (2017) A unified approach to interpreting model predictions. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds) Advances in Neural Information Processing Systems 30, Curran Associates, Inc., pp 4765–4774
Löwe H, Scholbeck CA, Heumann C, Bischl B, Casalicchio G (2023) fmeffects: An R package for forward marginal effects. arXiv e-prints arXiv:2310.02008
Mize TD, Doan L, Long JS (2019) A general framework for comparing predictions and marginal effects across models. Sociological Methodology 49(1):152–189
Molnar C (2022) Interpretable Machine Learning, 2nd edn. https://christophm.github.io/interpretable-ml-book
Molnar C, Casalicchio G, Bischl B (2020) Quantifying model complexity via functional decomposition for better post-hoc interpretability. In: Cellier P, Driessens K (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Communications in Computer and Information Science, vol 1167, Springer, Cham
Molnar C, König G, Herbinger J, Freiesleben T, Dandl S, Scholbeck CA, Casalicchio G, Grosse-Wentrup M, Bischl B (2022) General pitfalls of model-agnostic interpretation methods for machine learning models. In: Holzinger A, Goebel R, Fong R, Moon T, Müller KR, Samek W (eds) xxAI - Beyond Explainable AI. xxAI 2020. Lecture Notes in Computer Science, vol 13200, Springer, Cham
Morris MD (1991) Factorial sampling plans for preliminary computational experiments. Technometrics 33(2):161–174
Mullahy J (2017) Marginal effects in multivariate probit models. Empirical economics 53(2):447–461
Munson MA, Kegelmeyer WP (2013) Builtin vs. auxiliary detection of extrapolation risk. Tech. rep., Sandia National Laboratories, Albuquerque, New Mexico and Livermore, California
Norouzi M, Collins MD, Johnson M, Fleet DJ, Kohli P (2015) Efficient non-greedy optimization of decision trees. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, MIT Press, Cambridge, MA, USA, NIPS'15, p 1729-1737
Norton EC, Dowd BE, Maciejewski ML (2019) Marginal effects-quantifying the effect of changes in risk factors in logistic regression models. JAMA 321(13):1304–1305
Onukwugha E, Bergtold J, Jain R (2015) A primer on marginal effects-part II: Health services research applications. PharmacoEconomics 33(2):97–103
Philipp M, Zeileis A, Strobl C (2016) A toolkit for stability assessment of tree-based learners. In: Proceedings of COMPSTAT 2016 - 22nd International Conference on Computational Statistics, The International Statistical Institute/International Association for Statistical Computing, p 315-325
Quinlan JR (1993) C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo, California
Ramsey SM, Bergtold JS (2021) Examining inferences from neural network estimators of binary choice processes: Marginal effects, and willingness-to-pay. Computational Economics 58(4):1137–1165
Razavi S, Gupta HV (2016) A new framework for comprehensive, robust, and efficient global sensitivity analysis: 1. Theory. Water Resources Research 52(1):423–439
Razavi S, Jakeman A, Saltelli A, Prieur C, Iooss B, Borgonovo E, Plischke E, Lo Piano S, Iwanaga T, Becker W, Tarantola S, Guillaume JH, Jakeman J, Gupta H, Melillo N, Rabitti G, Chabridon V, Duan Q, Sun X, Smith S, Sheikholeslami R, Hosseini N, Asadzadeh M, Puy A, Kucherenko S, Maier HR (2021) The future of sensitivity analysis: An essential discipline for systems modeling and policy support. Environmental Modelling and Software 137:104954
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, Association for Computing Machinery, New York, NY, USA, KDD '16, p 1135-1144
Saltelli A, Ratto M, Andres T, Campolongo F, Cariboni J, Gatelli D, Saisana M, Tarantola S (2008) Global Sensitivity Analysis: The Primer. John Wiley & Sons, Ltd

Scholbeck CA, Molnar C, Heumann C, Bischl B, Casalicchio G (2020) Sampling, intervention, prediction, aggregation: A generalized framework for model-agnostic interpretations. In: Cellier P, Driessens K (eds) Machine Learning and Knowledge Discovery in Databases. ECML PKDD 2019. Communications in Computer and Information Science, vol 1167, Springer, Cham

Seibold H, Zeileis A, Hothorn T (2016) Model-based recursive partitioning for subgroup analyses. The International Journal of Biostatistics 12(1):45–63

Slack D, Hilgard S, Jia E, Singh S, 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, Association for Computing Machinery, New York, NY, USA, AIES '20, p 180-186

Sobol I, Kucherenko S (2010) Derivative based global sensitivity measures. Procedia - Social and Behavioral Sciences 2(6):7745 – 7746, Sixth International Conference on Sensitivity Analysis of Model Output

Stachl C, Hilbert S, Au JQ, Buschek D, De Luca A, Bischl B, Hussmann H, Bühner M (2017) Personality traits predict smartphone usage. European Journal of Personality 31(6):701–722

StataCorp, (2023) Stata Statistical Software: Release 18. StataCorp LLC, College Station, TX

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

Turney P (1995) Technical note: Bias and the quantification of stability. Machine Learning 20(1):23–33

Wachter S, Mittelstadt B, Russell C (2018) Counterfactual explanations without opening the black box: Automated decisions and the GDPR. Harvard Journal of Law and Technology 31(2):841–887

Williams R (2012) Using the margins command to estimate and interpret adjusted predictions and marginal effects. Stata Journal (24) 12(2):308–331

Zeileis A, Hothorn T, Hornik K (2008) Model-based recursive partitioning. Journal of Computational and Graphical Statistics 17(2):492–514

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 Behaviour and Society 20:22–35

Zhou Y, Zhou Z, Hooker G (2023) Approximation trees: Statistical reproducibility in model distillation. Data Mining and Knowledge Discovery. https://doi.org/10.1007/s10618-022-00907-3

Zhou Z, Hooker G, Wang F (2021) S-LIME: Stabilized-lime for model explanation. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA, 2429-2438

9 fmeffects: An R Package for Forward Marginal Effects

Contributing Paper

Löwe, H., Scholbeck, C. A., Heumann, C., Bischl, B., and Casalicchio, G. (2023). "fmeffects: An R Package for Forward Marginal Effects". In: arXiv: `2310.02008 [cs.LG]`

Declaration of Contributions

C.A. Scholbeck and H. Löwe share the first authorship of this paper. The paper builds upon the bachelor thesis of H. Löwe[1], which was supervised by C.A. Scholbeck and C. Heumann under close academic guidance and supervision. C.A. Scholbeck jointly conceptualized the software design with H. Löwe and C. Heumann and provided continuous support throughout the development process. The software package as described in the paper extends the bachelor thesis in multiple directions: The package functionality was extended by AMEs, a model summary output, binary classification, and the support for `tidymodels`; the package was renamed as `fmeffects`; a comprehensive vignette describing the package was added to the repository; and the package was published on the *Comprehensive R Archive Network*. C.A. Scholbeck contributed minor modifications to the code, wrote the paper, and revised it according to the feedback from his co-authors.

H. Löwe wrote and documented the entire software package up to minor modifications and created the initial applied example used in the paper. C. Heumann, B. Bischl, and G. Casalicchio reviewed the software and suggested several notable modifications. All authors assisted in revising the paper.

[1] Löwe, Holger (2022): fme – An R Package for Forward Marginal Effects. Bachelor Thesis, Ludwig-Maximilians-Universität München

fmeffects: An R Package for Forward Marginal Effects

Holger Löwe[1], *Christian A. Scholbeck*[1], *Christian Heumann*, *Bernd Bischl and Giuseppe Casalicchio*

Abstract

Forward marginal effects have recently been introduced as a versatile and effective model-agnostic interpretation method particularly suited for non-linear and non-parametric prediction models. They provide comprehensible model explanations of the form: if we change feature values by a pre-specified step size, what is the change in the predicted outcome? We present the R package **fmeffects**, the first software implementation of the theory surrounding forward marginal effects. The relevant theoretical background, package functionality and handling, as well as the software design and options for future extensions are discussed in this paper.

Introduction

A growing number of disciplines are adopting black box machine learning (ML) models to make predictions, including medicine (Rajkomar et al., 2019; Boulesteix et al., 2020), psychology (Dwyer et al., 2018), economics (Mullainathan and Spiess, 2017; Athey and Imbens, 2019), or the earth sciences (Dueben and Bauer, 2018). Although one can often observe a superior predictive performance of black box models (such as neural networks, gradient boosting, random forests, or support vector machines) over intrinsically interpretable models (such as generalized linear or additive models), their lack of transparency or interpretability is considered a major drawback (Breiman, 2001). This has been a major driver in the development of model-agnostic explanation techniques, which are often referred to by the umbrella terms of interpretable ML (Molnar, 2022) or explainable artificial intelligence (Kamath and Liu, 2021).

Marginal effects (MEs) (Williams, 2012) have been a mainstay of model interpretations in many applied fields such as econometrics (Greene, 2019), psychology (McCabe et al., 2022), or medical research (Onukwugha et al., 2015). MEs explain the effect of features on the predicted outcome in terms of derivatives w.r.t. a feature or forward differences in prediction. They are typically averaged to an average marginal effect (AME) for an entire data set, which serves as a global (scalar-valued) feature effect measure (Bartus, 2005). To explain feature effects for non-linear models, Scholbeck et al. (2024) introduced a unified definition of forward marginal effects (FMEs), a non-linearity measure (NLM) for FMEs, and the conditional average marginal effect (cAME). The NLM is an auxiliary model diagnostic to avoid interpreting local changes in prediction as linear effects. The cAME aims to describe the model via regional FME averages for subgroups with similar FMEs, which can, for instance, be found by recursive partitioning (RP). FMEs, therefore, represent a means to explain models on a local, regional, and global level.

Contributions: We present the R package **fmeffects**, the first software implementation of the theory surrounding FMEs, including the NLM and the cAME. The user interface only requires a pre-trained model and an evaluation data set. The package is designed according to modular principles, making it simple to maintain and extend. This paper introduces the relevant theoretical background of FMEs, demonstrates the usage of the package in the context of a practical use case, and explains the software design.

Background on forward marginal effects

FMEs can be used for model explanations on the local, regional (also referred to as semi-global), and global level. These differ with respect to the region of the feature space that the explanation refers to. The local level explains a model/prediction for single observations, the regional level for a certain subspace (or subgroups of observations), and the global level for the entire feature space. Increasing the scope of the explanation requires increasing amounts of aggregations of local explanations (see the illustration by Scholbeck et al. (2020) of aggregations of local explanations to global ones for various methods). This can be problematic for non-parametric models where local explanations can be highly heterogeneous due to non-linear effects or interactions.

[1]H. Löwe and C.A. Scholbeck contributed equally.

Notation

Let $\widehat{f} : \mathcal{X} \to \mathbb{R}$ be the prediction function of a learned model where $\mathcal{X} \subset \mathbb{R}^p$ denotes the feature space. While our definition naturally covers regression models, for classification models, we assume that $\widehat{f}$ returns the score or probability for a predefined class of interest. A subspace of the feature space is denoted by $\mathcal{X}_{[\,]} \subseteq \mathcal{X}$. The random feature vector is denoted by[1] $\boldsymbol{X} = (X_1, \ldots, X_p)$. Observations are denoted by $\boldsymbol{x} = (x_1, \ldots, x_p) \in \mathcal{X}$. A set of feature indices is denoted by $S \subseteq \{1, \ldots, p\}$. We often index (random) vectors as $\boldsymbol{x}_S$ or $\boldsymbol{X}_S$. We denote set complements by $-S = \{1, \ldots, p\} \setminus S$. With slight abuse of notation, we represent the partitioning of a vector into two arbitrary but disjoint groups by $\boldsymbol{x} = (\boldsymbol{x}_S, \boldsymbol{x}_{-S})$, regardless of the order of features. For a single feature of interest, the set S is replaced by an integer index j. We usually assume an evaluation data set $\mathcal{D} = \left(\boldsymbol{x}^{(i)}\right)_{i=1}^{n}$, with $\boldsymbol{x}^{(i)} \in \mathcal{X}$, which may consist of both training and test data.

Forward marginal effects

The FME can be considered a basic, local unit of interpretation. Given an observation $\boldsymbol{x}$, it tells us how the prediction changes if we change a subset of feature values $\boldsymbol{x}_S$ by a vector of step sizes $\boldsymbol{h}_S$.

$$\text{FME}_{\boldsymbol{x}, \boldsymbol{h}_S} = \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S}) - \widehat{f}(\boldsymbol{x}) \qquad \text{for continuous features } \boldsymbol{x}_S$$

Scholbeck et al. (2024) introduced an observation-specific categorical FME whose definition is congruent with the FME for continuous features. The categorical FME corresponds to the change in prediction when replacing x_j by the reference category c_j:

$$\text{FME}_{\boldsymbol{x}, c_j} = \widehat{f}(c_j, \boldsymbol{x}_{-j}) - \widehat{f}(\boldsymbol{x}) \qquad \text{for categorical } x_j$$

Note that this definition of a categorical ME differs from the one that is typically found in fields like econometrics (Williams, 2012), where we set x_j to a reference category for all observations and then record the change in prediction resulting from changing the reference category to another category.

Furthermore, it is common to globally average MEs to an average marginal effect (AME) to estimate the expected local effect. For FMEs, this corresponds to:

$$\begin{aligned} \text{AME}_{\mathcal{D}, \boldsymbol{h}_S} &= \widehat{\mathbb{E}_{\boldsymbol{X}}\left[\text{FME}_{\boldsymbol{X}, \boldsymbol{h}_S}\right]} \\ &= \frac{1}{n} \sum_{i=1}^{n} \left[\widehat{f}\left(\boldsymbol{x}_S^{(i)} + \boldsymbol{h}_S, \boldsymbol{x}_{-S}^{(i)}\right) - \widehat{f}\left(\boldsymbol{x}^{(i)}\right)\right] \end{aligned} \tag{1}$$

Note that for categorical feature changes and observations where $x_j = c_j$, the FME equals 0. In the **fmeffects** package, the categorical AME only consists of observations whose observed feature value differs from the selected category. This approach is motivated by our goal to explain the effects of *changing feature values* on the predicted outcome. For instance, in Fig. 11, we demonstrate the effect of rainfall on the daily number of bike rentals in Washington D.C. by switching each non-rainy day's precipitation status to rainfall. Considering all observations, including rainy days, would obfuscate the interpretation we desire from our model. However, it is important to remember that every AME comprises a different set of points.

Step size selection

The selection of step sizes is determined by contextual and data-related considerations (Scholbeck et al., 2024). First, the FME allows us to investigate the model according to specific research questions. For instance, we might be interested in the effects of a specific change in a patient's body weight on the predicted individual disease risk. Often, we are interested in an interpretable or intuitive step size. In the case of body weight, typically expressed in kilograms, we could use a 1kg change (for instance, instead of 1g) as a natural increment. Without contextual information, we could use a unit change as a reasonable default; or dispersion-based measures such as one standard deviation, percentages of the interquartile range, or the mean/median absolute deviation.

Non-linearity measure

For continuous features, we can consider $\boldsymbol{x}_S + \boldsymbol{h}_S$ a continuous transition of feature values. The associated change in prediction may be misinterpreted as a linear effect. This is counteracted by the

[1] Bold letters denote vectors.

NLM, which corresponds to a continuous coefficient of determination R^2 between the prediction function and the linear secant that intersects $\boldsymbol{x}$ and $(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S})$ (see Fig. 1). The continuous transition through the feature space is first parameterized as a fraction $t \in [0, 1]$ of the multivariate step size $\boldsymbol{h}_S$:

$$\gamma(t) = \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} + t \cdot \begin{pmatrix} h_1 \\ \vdots \\ h_s \\ 0 \\ \vdots \\ 0 \end{pmatrix} \quad , \quad t \in [0, 1]$$

The value of the linear secant $g_{\boldsymbol{x},\boldsymbol{h}_S}(t)$ corresponds to:

$$g_{\boldsymbol{x},\boldsymbol{h}_S}(t) = \begin{pmatrix} x_1 + t \cdot h_1 \\ \vdots \\ x_s + t \cdot h_s \\ \vdots \\ x_p \\ \widehat{f}(\boldsymbol{x}) + t \cdot \text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} \end{pmatrix}$$

The mean prediction $\widehat{f}_{\text{mean}}$ on the interval $t \in [0, 1]$ is given by:

$$\begin{aligned} \widehat{f}_{\text{mean}} &= \frac{\int_0^1 \widehat{f}(\gamma(t)) \left|\left| \frac{\partial \gamma(t)}{\partial t} \right|\right|_2 dt}{\int_0^1 \left|\left| \frac{\partial \gamma(t)}{\partial t} \right|\right|_2 dt} \\ &= \int_0^1 \widehat{f}(\gamma(t)) \, dt \end{aligned}$$

The NLM compares the squared deviation between the prediction function and the linear secant to the squared deviation between the prediction function and the mean prediction:

$$\text{NLM}_{\boldsymbol{x},\boldsymbol{h}_S} = 1 - \frac{\int_0^1 \left(\widehat{f}(\gamma(t)) - g_{\boldsymbol{x},\boldsymbol{h}_S}(t)\right)^2 \left|\left| \frac{\partial \gamma(t)}{\partial t} \right|\right|_2 dt}{\int_0^1 \left(\widehat{f}(\gamma(t)) - \widehat{f}_{\text{mean}}\right)^2 \left|\left| \frac{\partial \gamma(t)}{\partial t} \right|\right|_2 dt} \quad \in (-\infty, 1]$$

Fig. 2 illustrates the setting for multivariate feature changes. The NLM can be approximated via numerical integration, e.g., via Simpson's rule.

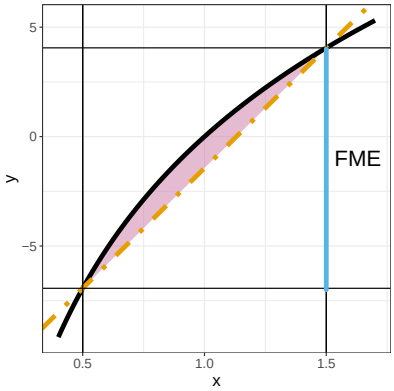


Figure 1: Illustration by Scholbeck et al. (2024) of a univariate FME (blue) given the prediction function (black) and linear secant (orange, dashed). The NLM indicates how well the secant can explain the prediction function (inversely proportional to the purple area) compared to how well the most uninformative baseline model (the average prediction) can explain the prediction function.

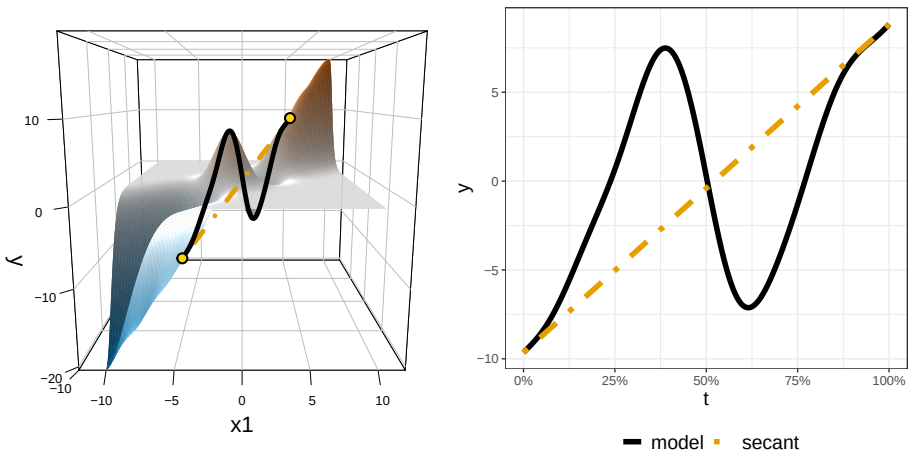


Figure 2: Illustration of the multivariate NLM by Scholbeck et al. (2024). **Left:** An exemplary bivariate prediction function and two points to compute an FME. Consider an observation $\boldsymbol{x} = (-5, -5)$ and step size vector $\boldsymbol{h}_S = (10, 10)$. We create the shortest path through the feature space to reach the point (5, 5), which consists of directly proportional changes in both features. Above the path, we see the linear secant (orange, dashed) and the non-linear prediction function (black). **Right:** The multivariate change in feature values can be parameterized as a percentage t of the step size $\boldsymbol{h}_S$. The deviation between the prediction function and the linear secant, as well as the deviation between the prediction function and mean prediction, both correspond to a line integral.

The NLM indicates how well the linear secant can explain the prediction function, compared to the baseline model of using the mean prediction. A value of 1 indicates perfect linearity, where the linear secant is identical to the prediction function. For a value of 0, the mean prediction can explain the prediction function to the same degree as the secant. For negative values, the mean prediction better explains the prediction function than the linear secant (severe non-linearity).

It is, therefore, easiest to interpret FMEs with NLM values close to 1. Although every FME always represents the exact change in prediction, an FME with a low NLM value does not fully describe the behavior of the model in that specific locality. In contrast, an FME with an NLM close to 1 is a sufficient descriptor of the (linear) model behavior. In other words, the NLM serves as an auxiliary diagnostic tool, indicating trust in how well the FME describes the local change in prediction.

Conditional average marginal effect

To receive a global model explanation akin to a beta coefficient in linear models, local FMEs can be averaged to the AME. Mehrabi et al. (2021) define an *aggregation bias* as drawing false conclusions about individuals from observing the entire population. Given a data set $\mathcal{D}$, the conditional average marginal effect (cAME) estimator applies to a subgroup of $n_{[\,]}$ observations, denoted by $\mathcal{D}_{[\,]}$:

$$\begin{aligned} \text{cAME}_{\mathcal{D}_{[\,]},\boldsymbol{h}_S} &= \widehat{\mathbb{E}_{\boldsymbol{X}_{[\,]}}\left[\text{FME}_{\boldsymbol{X}_{[\,]},\boldsymbol{h}_S}\right]} \\ &= \frac{1}{n_{[\,]}} \sum_{i:\boldsymbol{x}^{(i)} \in \mathcal{D}_{[\,]}} \left[\widehat{f}\left(\boldsymbol{x}_S^{(i)} + \boldsymbol{h}_S, \boldsymbol{x}_{-S}^{(i)}\right) - \widehat{f}\left(\boldsymbol{x}^{(i)}\right)\right] \end{aligned} \tag{2}$$

Although this estimator can be applied to arbitrary subgroups, we aim to find subgroups with cAMEs that counteract the aggregation bias. Desiderata for such subgroups include within-group effect homogeneity, between-group effect heterogeneity, full segmentation, non-congruence, confidence, and stability (Scholbeck et al., 2024). In other words, we aim to partition the data into subgroups that explain variability in the FMEs. A viable option to partition $\mathcal{D}$ is to run RP on $\mathcal{D}$ with FMEs as the target. For instance, in **fmeffects**, both **rpart** (Therneau and Atkinson, 2019) and `ctree()` from **partykit** (Hothorn and Zeileis, 2015) are supported to find subgroups.

Related work

Model-agnostic interpretations

The basic mechanism behind model-agnostic methods is to probe the model with different feature values, a methodology similar to a model sensitivity analysis (Scholbeck et al., 2020, 2023). The basis of explaining models is to determine the direction and magnitude of the effect of features on the predicted outcome (Casalicchio et al., 2019; Scholbeck et al., 2020, 2024). The individual conditional expectation (ICE) (Goldstein et al., 2015), partial dependence (PD) (Friedman, 2001), accumulated local effects (ALE) (Apley and Zhu, 2020), Shapley values (Štrumbelj and Kononenko, 2010; Lundberg and Lee, 2017; Covert et al., 2020) and local interpretable model-agnostic explanations (LIME) (Ribeiro et al., 2016) are some of the most popular model-agnostic explanation methods for ML models. Notably, counterfactual explanations (Wachter et al., 2018) represent the reverse of the FME, indicating the smallest necessary change in feature values to reach a targeted prediction.

FMEs complement the literature by allowing for a unique combination of local, regional, and global model explanations. Furthermore, while most methods (including the ICE, PD, ALE, or Shapley values) provide explanations in terms of prediction *levels*, FMEs provide explanations in terms of prediction *changes*. LIME is based on training a local and interpretable surrogate model whose coefficients can also provide an interpretation in terms of prediction changes. Scholbeck et al. (2024) highlighted differences between both approaches: notably, while surrogate models introduce additional uncertainty connected with the estimation of the surrogate, FMEs are motivated by the goal of stable and comprehensible model insight. Furthermore, locally estimated FMEs can be aggregated within subgroups and entire data sets for regional and global explanations. Around the same time, regional aggregations have also been introduced for ICE curves, for example (Britton, 2019; Herbinger et al., 2022; Molnar et al., 2024).

Relationship between individual conditional expectation and forward marginal effect

Scholbeck et al. (2024) illustrated a relationship between the ICE / PD and the FME / AME. In general, the FME can be seen as the difference between two locations on an ICE. The AME corresponds to the difference between two locations on the PD only for a function that is linear in the feature of interest. Therefore, the following relationship between the ICE and FME is worth noting here. The ICE can be considered a one-way sensitivity function that indicates the effects of varying a set of features indexed by S while the remaining ones are kept constant:

$$\text{ICE}_{\boldsymbol{x},S}(\boldsymbol{x}_S^*) = \widehat{f}(\boldsymbol{x}_S^*, \boldsymbol{x}_{-S})$$

For an instance $\boldsymbol{x}$, the prediction after increasing $\boldsymbol{x}_S$ by $\boldsymbol{h}_S$ is also a value of the ICE:

$$\begin{aligned}\text{FME}_{\boldsymbol{x},\boldsymbol{h}_S} &= \widehat{f}(\boldsymbol{x}_S + \boldsymbol{h}_S, \boldsymbol{x}_{-S}) - \widehat{f}(\boldsymbol{x}) \\ &= \text{ICE}_{\boldsymbol{x},S}(\boldsymbol{x}_S + \boldsymbol{h}_S) - \text{ICE}_{\boldsymbol{x},S}(\boldsymbol{x}_S)\end{aligned}$$

Related work on marginal effects

MEs have a long history in applied statistics and the Stata programming language (StataCorp, 2023). Initially implemented by Bartus (2005), the `margins()` command is now fully integrated into Stata and provides comprehensive capabilities for various computations and visualizations of statistical models such as (generalized) linear models (Williams, 2012). MEs are typically defined in terms of derivatives of the model w.r.t. a feature. For instance, this variant is the default approach to interpret models in econometrics (Greene, 2019). The FME is the less commonly used definition (Scholbeck et al., 2024; Mize et al., 2019). Note that—in contrast to forward differences—derivatives are not suitable to explain piecewise constant prediction functions such as tree-based models.

In recent years, MEs have gained traction in the R community. The R package **margins** (Leeper, 2018) was the first port of Stata's `margins()` command to R. Other packages related to MEs include **ggeffects** (Lüdecke, 2018) and **marginaleffects** (Arel-Bundock, 2023). In particular, **marginaleffects** can also return FMEs (although under different terminology). Our package, **fmeffects**, mainly differs from **marginaleffects** in two aspects:

Implementing new theory surrounding FMEs: The **fmeffects** package is the first software implementation of the theory surrounding model-agnostic FMEs as introduced by Scholbeck et al. (2024). Although packages such as **marginaleffects** support the computation of FMEs and other quantities, **fmeffects** is specifically designed for FMEs with unique features such as implementations of the NLM, the cAME via RP, and novel visualization methods.

Model-agnostic black box interpretations: It follows that **fmeffects** is targeted at model-agnostic explanations of non-linear or intransparent models. Whereas existing theory on MEs (and packages such as **marginaleffects**) focuses on classical statistical modeling in combination with statistical inference (see, for instance, Breiman (2001) comparing statistical modeling culture with ML), FMEs (and thus **fmeffects**) are comparable to methods and software from the literature on interpretable ML such as the ICE, PD, ALE, or LIME. This does not imply that **marginaleffects** cannot be used for black box interpretations. As mentioned in the previous point, it also supports the computation of FMEs, e.g., in combination with **mlr3**, but the focus of **fmeffects** lies on the interpretation of black box models through a specialized and targeted range of novel capabilities.

Advantages and limitations of forward marginal effects

Advantages

Although the ICE and the FME are closely related, the latter provides several novel ways to generate insights into the model:

- **Univariate changes in feature values:** FMEs are comparable to ICE curves for univariate changes in feature values. In certain scenarios, however, they may provide more comprehensible visualizations of effects for individual instances (see Fig. 4 for an example).
- **Bivariate changes in feature values:** The ICE and PD also provide insight into the sensitivity of the model prediction for variations in two features, which is visualized as a heatmap (see Fig. 7). However, it is difficult to visually compare the ICE of many different observations (which correspond to heatmaps as well). Although the ICE provides insight into a larger variation in feature values, while the FME only considers a single tuple of changes in feature values, bivariate FMEs can be easily compared visually (see Fig. 6).
- **Higher-order changes in feature values:** If we evaluate the sensitivity of the prediction for changes in more than two feature values, virtually every visualization method breaks down. In this case, FMEs still provide comprehensible model explanations that can be aggregated in various ways (see Fig. 10).
- **Local fidelity assessment:** The locally restricted change in feature values for the FME facilitates evaluations of the fidelity of the model explanation (e.g., via the NLM). In other words, the NLM allows us to describe how well the FME summarizes the local shape of the prediction function in a single value. See Fig. 8 for a visualization of NLM values for different observations.
- **Comprehensible regional explanations:** Although regional explanations have been first proposed in the context of grouping ICE curves (Herbinger et al., 2022; Britton, 2019), they more easily apply to scalar model explanations such as FMEs. Essentially, a regional model explanation represents a group of observations or a subspace of the feature space where model explanations are relatively homogeneous. Such groupings are easily achievable via RP or other techniques that do not require functional target values such as ICEs.
- **Avoiding extrapolation:** The ICE is computed on the entire feature range (see, e.g., Fig. 4), which is likely to result in model extrapolations. By its nature, the FME is typically used with small step sizes relative to the feature range, which naturally avoids model extrapolations.

Limitations

- **Step size selection:** The step size fundamentally influences effects and the model interpretation. Although FMEs for different step sizes can be computed and visualized in an exploratory manner, some level of prior reasoning about what step sizes to use is recommended.
- **Decision tree instability for cAME:** Although not a shortcoming of the FME itself, subgroups found by RP to compute cAMEs are subject to a high variance. This may be counteracted by stabilizing the split search, e.g., by considering statistical significance of tree splits or resorting to different algorithms to find subgroups.
- **Non-linearity assessment for proportional feature changes:** For multi-dimensional feature changes, the NLM only considers equally proportional changes in all features.

On causal interpretations and avoiding model extrapolations

Note that model-agnostic techniques, including FMEs, explain associations between the target and the features within the model. In the absence of additional assumptions, such associations cannot be interpreted as causes and effects (Molnar et al., 2022). For instance, increasing the value of a feature x_1 may always be accompanied by an increase in the target, but it may be the target y that causes x_1 to increase. Another typical scenario is the presence of confounding factors that influence both y and x_1. Finally, x_1 may only (or also) influence a mediator x_2, which in turn influences y.

This does not, however, make model interpretations obsolete. More importantly, as highlighted by Adadi and Berrada (2018), model interpretations can be used to gain knowledge, debug, audit, or justify the model and its predictions. Throughout this paper, we will model the effects of environmental influences on the number of daily bike rentals in Washington, D.C. For our estimated model, a drop in humidity by 10 percentage points has a considerable effect on the predicted number of daily bike rentals (see Fig. 5). This effect cannot be assumed to be causal, as humidity is physically influenced by the outside temperature, which will also affect people's choice to rent a bike. Here, temperature is a confounder that influences both humidity and daily bike rentals. However, the business renting out bikes can still use the associations found by a model with a good predictive performance to control the optimal number of bikes at their disposal. This is conditional on the model's ability to accurately predict the target for the given feature vector, requiring us to avoid model extrapolations, which correspond to predictions within areas of the feature space where the model has not seen much or any training data. This issue is closely linked to the multivariate distribution of the training data; in our example, a change in humidity is likely to be accompanied by a change in temperature as well, which we somewhat circumvent (depending on the magnitude of the step size) when making isolated changes to humidity. One may disregard this issue and deliberately predict in areas of the feature space the model has not seen during training. The resulting FMEs will still be valid model descriptions but, as explained above, they are likely to be bad descriptions of the data generating process.

Model extrapolations negatively impact many model-agnostic interpretation methods (Hooker, 2004b,a, 2007; Hooker et al., 2021; Molnar et al., 2022). For example, Apley and Zhu (2020) demonstrated how PD plots suffer from extrapolation issues and introduced ALE plots as a solution to this problem. Scholbeck et al. (2024) illustrated the perils of model extrapolations for FMEs specifically and discussed possible options. One option in particular is also implemented in **fmeffects**: points outside the multivariate envelope (meaning the Cartesian product of all observed feature ranges) of the training data can be excluded from the analysis. This directly relates to the selection of small step sizes relative to the feature range, as large step sizes will result in a point falling outside the envelope.

When using extrapolation prevention methods, note that we consider different sets of points for different step sizes, which differs from the usage of MEs in other contexts (see, for instance, the package **marginaleffects** for a comparison). The exclusion of points only impacts aggregations of FMEs, i.e., the cAME and AME. As discussed in the section on Forward marginal effects, this also affects the computation of categorical AMEs. In Eq. (1) and Eq. (2), the AME and cAME are formulated as estimators of the expected global or regional (concerning a subspace) effects. The fewer observations we are considering for an average, the larger the variance of the estimate.

User interface and package handling

Local explanations

The `fme()` function is the central user interface. It mainly requires a pre-trained model and a data set (see section Design and options for extensions for details). Further control parameters include a list of features and step sizes, whether to compute NLM values for each FME, and an extrapolation detection method. The `fme()` function initiates the construction and computations of a `ForwardMarginalEffect` object without requiring the user to know **R6** (Chang, 2021) syntax.

For this use case, we train a random forest from the **ranger** package (Wright and Ziegler, 2017) on the bike sharing data set (Fanaee-T, 2013) using **mlr3**. Note that models trained via **tidymodels** and **caret** are also supported, as well as models trained via `lm()`, `glm()`, and `gam()`. We aim to predict and explain the daily bike rental demand in Washington, D.C., based on features such as the outside temperature, wind speed, or humidity. We first train the model:

```
> library(fmeffects)
> data(bikes, package = "fmeffects")
> library(mlr3verse)
> forest = lrn("regr.ranger")
```

```
> task = as_task_regr(x = bikes, id = "bikes", target = "count")
> forest$train(task)
```

Then, we simply pass the trained model, evaluation data, and remaining parameters to the `fme()` function. It returns a `ForwardMarginalEffect` object, which can be analyzed via `summary()` and visualized via `plot()` (see Fig. 3). Here, the outside temperature is raised by 5 degrees Celsius ceteris paribus. To avoid overplotting values, each hexagon represents a local average of FMEs. Users can easily access the data used by all plot functions to implement their own visualizations.

Let us single out the observation with the largest associated FME. This observation corresponds to a single day with a recorded temperature of 8 degrees Celsius. Increasing the temperature by 5 degrees Celsius on this particular day results in 2699 additional predicted bike rentals. We plot such model explanations for the entire data set and average FMEs to receive a global model explanation. The AME—the global average of FMEs—is 307: an increase in temperature by 5 degrees Celsius results in an average increase of 307 predicted daily bike rentals.

```
> effects.univariate.temp = fme(
+   model = forest,
+   data = bikes,
+   features = list("temp" = 5),
+   ep.method = "envelope")

> summary(effects.univariate.temp)

Forward Marginal Effects Object

Step type:
  numerical

Features & step lengths:
  temp, 5

Extrapolation point detection:
  envelope, EPs: 48 of 731 obs. (7 %)

Average Marginal Effect (AME):
  307.3275

> plot(effects.univariate.temp)
```

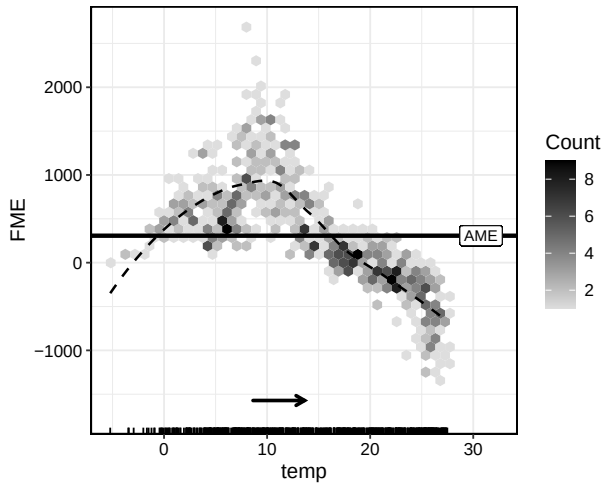


Figure 3: Plot of univariate FMEs for feature 'temp' and step size 5. Each hexagon represents a local FME average. The horizontal value represents the observed feature value of 'temp'. Each observation's 'temp' value is moved according to the arrow's direction and length. The vertical value of each hexagon indicates the FME value associated with that feature change. The horizontal bar indicates the AME. The shade of the hexagon implies how many observations it contains. A smoothing function facilitates interpretations by modeling an approximate pattern of FMEs across the feature range.

Let us take a moment to compare the FME plot with the combined ICE and PD plot generated by the R package **iml** (Molnar et al., 2018) (see Fig. 4). This is one of the most popular and established model-agnostic ways to interpret predictive models (Molnar, 2022). The ICE is a local model explanation and represents the prediction for an observation where only the features of interest are varied (in this case, only 'temp'). The PD is the average of ICEs (in the univariate case, the vertical average) and indicates the global, average prediction when a subset of features is varied for all observations. Although we can see a rough trajectory of the feature influence on local and average predictions, it is difficult to pinpoint the exact effects of changing 'temp' on the prediction for single observations. Furthermore, ICE curves are more likely to be subject to model extrapolations, a result of predicting in areas where the model was not trained on a sufficient amount of data.

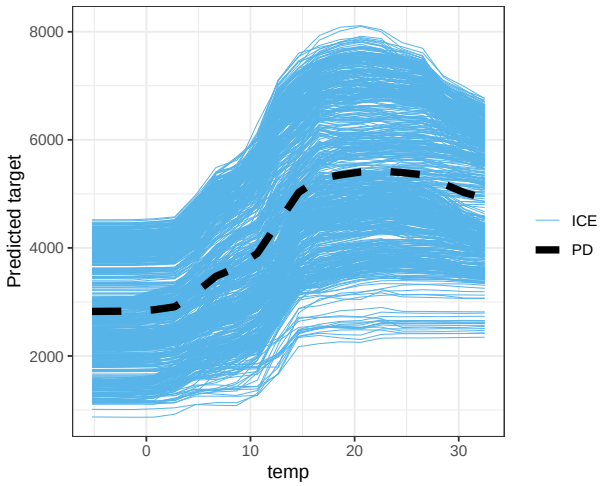


Figure 4: An ICE and PD plot for feature 'temp' generated by the R package **iml**. Each solid blue curve (an ICE) represents predictions for a single instance while only 'temp' varies. The dashed black curve (the PD) is the vertical average of ICEs and represents the average, isolated influence of 'temp'.

FMEs allow for positive or negative step sizes. For instance, let us investigate the effects of an isolated drop in humidity by 10 percentage points. We can observe an AME of 108 additional predicted bike rentals a day. Individual effects tend to be larger the higher the humidity on that particular day.

```
> effects.univariate.humidity = fme(
+   model = forest,
+   data = bikes,
+   features = list("humidity" = -0.1),
+   ep.method = "envelope")

> summary(effects.univariate.humidity)

Forward Marginal Effects Object

Step type:
  numerical

Features & step lengths:
  humidity, -0.1

Extrapolation point detection:
  envelope, EPs: 1 of 731 obs. (0 %)

Average Marginal Effect (AME):
  108.0477

> plot(effects.univariate.humidity)
```

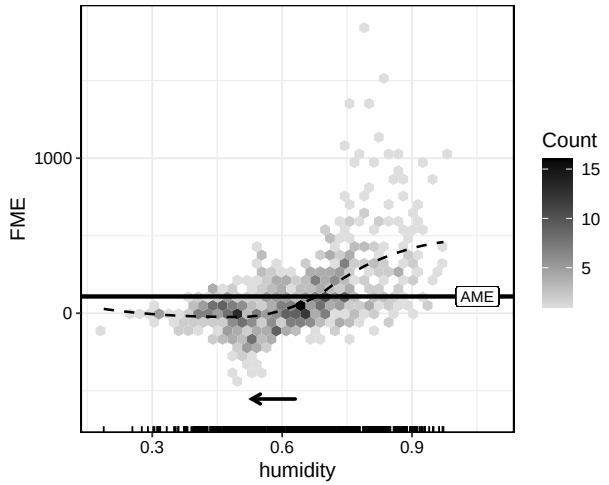


Figure 5: Univariate FMEs for a drop in humidity by 10 percentage points. Especially for high humidity values, the drop results in a considerable increase in predicted daily bike rentals.

In many applications, we are interested in interactions of features on the prediction. Until now, we only analyzed the univariate effects of 'temp' and 'humidity' on the predicted amount of bike rentals. However, potential interactions between features may exist. We evaluate an increase in temperature by 5 degrees Celsius and a simultaneous drop in humidity by 10 percentage points (see Fig. 6). For a bivariate change in feature values, the two arrows depict the direction and magnitude of the feature change in the respective variable. As in the univariate case, we plot local averages within hexagons to avoid overplotting values. The location of the hexagon is determined by the observations' observed feature values in the provided data set. Its color indicates the FME associated with the bivariate feature change. An increase in the outside temperature by 5 degrees Celsius and a simultaneous drop in humidity by 10 percentage points is associated with an AME of 414. The univariate AMEs roughly add up to the bivariate AME, indicating that, on average, there is no additional interaction between both feature changes on the prediction.

```
> effects.bivariate = fme(
+   model = forest,
+   data = bikes,
+   features = list("temp" = 5, "humidity" = -0.1),
+   ep.method = "envelope")

> summary(effects.bivariate)

Forward Marginal Effects Object

Step type:
  numerical

Features & step lengths:
  temp, 5
  humidity, -0.1

Extrapolation point detection:
  envelope, EPs: 49 of 731 obs. (7 %)

Average Marginal Effect (AME):
  413.6163

> plot(effects.bivariate)
```

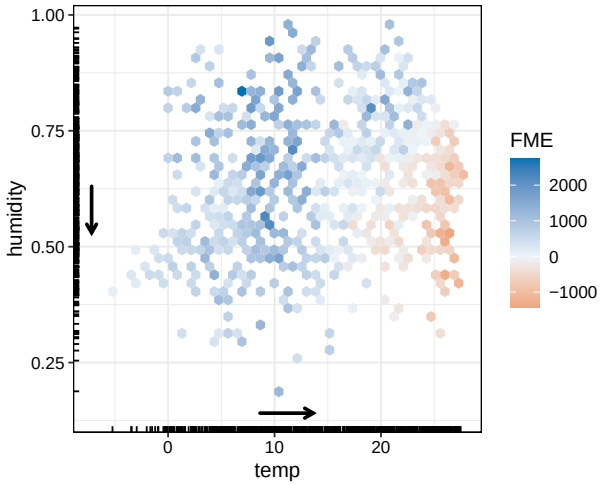


Figure 6: Visualizing bivariate FMEs for an increase in 'temp' by 5 degrees Celsius and a simultaneous drop in 'humidity' by 10 percentage points. FMEs are highly heterogeneous. We can see mostly positive effects, especially for observations with combinations of medium 'temp' and 'humidity' values.

Let us repeat the same procedure as for univariate feature changes and compare the FME plot to an alternative option, the bivariate PD plot (see Fig. 7). As opposed to the novel visualization with FMEs, the PD plot only visualizes the average, global effect of changing both features on the predicted amount of bike rentals. It does not inform us about the distribution of observed feature values, thus not allowing us to evaluate the effects of increasing one feature and decreasing another simultaneously.

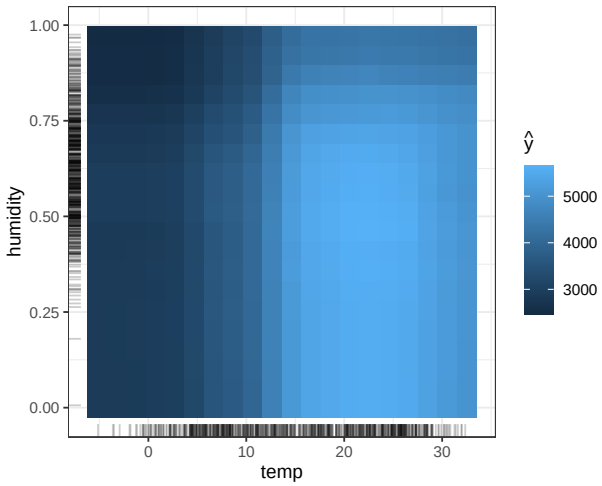


Figure 7: A bivariate PD plot (created via the R package **iml**), visualizing the global interaction between 'temp' and 'humidity' on the predicted amount of bike rentals. Plugging in medium to large values for 'temp' and low to medium values for 'humidity', ceteris paribus, results in more predicted bike rentals on average. As opposed to bivariate FMEs, we cannot investigate multiple local effects, nor can we see the actual distribution of observed feature values. As a result, we cannot evaluate the effects of increasing one feature and decreasing another simultaneously.

Let us now proceed to investigate non-linearity. Non-linearity can be visually assessed for ICE curves (see Fig. 4), but it is hard to quantify and would be somewhat meaningless for a large variation in the feature of interest. Furthermore, for bivariate or higher-dimensional changes in feature values, we lose any option for visual diagnoses of non-linearity. In contrast, the NLM can be computed for FMEs with continuous step sizes, regardless of dimensionality. The average non-linearity measure (ANLM) is 0.36, indicating that the linear secant, on average, is a bad descriptor of the FME.

```
> effects.bivariate.nlm = fme(
+   model = forest,
+   data = bikes,
+   features = list("temp" = 5, "humidity" = -0.1),
+   ep.method = "envelope",
+   compute.nlm = TRUE)

> effects.bivariate.nlm

Forward Marginal Effects Object

Features & step lengths:
  temp, 5
  humidity, -0.1

Average Marginal Effect (AME):
  413.6163

Average Non-Linearity Measure (ANLM):
  0.36

> plot(effects.bivariate.nlm, with.nlm = TRUE)
```

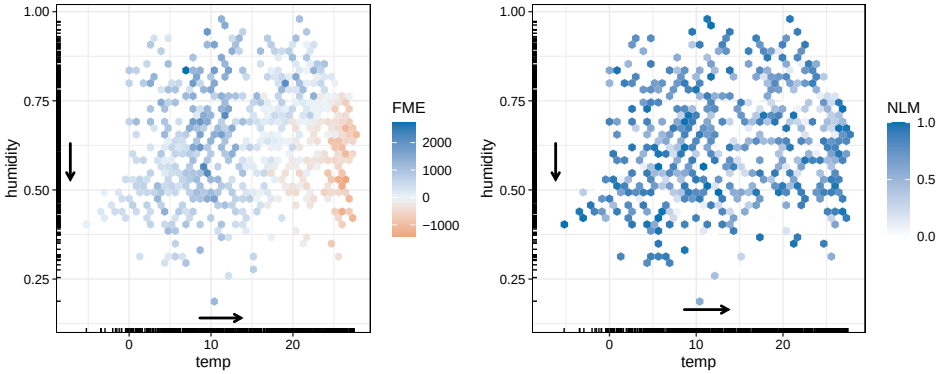


Figure 8: Adding NLM computations to the FME plot. Each hexagon in the left and right plots represents a local average of FME and NLM values, respectively.

Fig. 8 simply contrasts FME values with the corresponding NLM values. In this case, we can see both non-linear FMEs (whiter NLM) and linear FMEs (deep blue-colored NLM). We could now, for instance, focus on interpreting linear FMEs. All FMEs depicted in Fig. 9 have an NLM of 0.9 or higher, meaning that they almost fully describe the model prediction for proportional changes in 'temp' and 'humidity'.

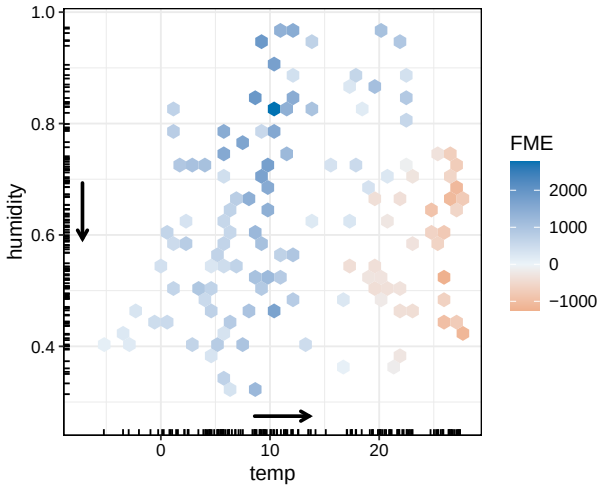


Figure 9: Visualizing FMEs with an NLM ≥ 0.9.

An advantage of FMEs is their ability to provide comprehensible model insight even when exploring higher-order feature changes. Let us factor in a third feature change, now simultaneously reducing windspeed by 5 miles per hour, and visualize the distribution of FME and NLM values. We can see that in addition to an increase in temperature and a decrease in humidity, a decrease in windspeed further boosts the average number of predicted daily bike rentals.

```
> effects.trivariate.nlm = fme(
+   model = forest,
+   data = bikes,
+   features = list("temp" = 5, "humidity" = -0.1, "windspeed" = -5),
+   ep.method = "envelope",
+   compute.nlm = TRUE)

> summary(effects.trivariate.nlm)

Forward Marginal Effects Object

Step type:
  numerical

Features & step lengths:
  temp, 5
  humidity, -0.1
  windspeed, -5

Extrapolation point detection:
  envelope, EPs: 117 of 731 obs. (16 %)

Average Marginal Effect (AME):
  537.7385

Average Non-Linearity Measure (ANLM):
  0.33

> plot(effects.trivariate.nlm, with.nlm = TRUE)
```

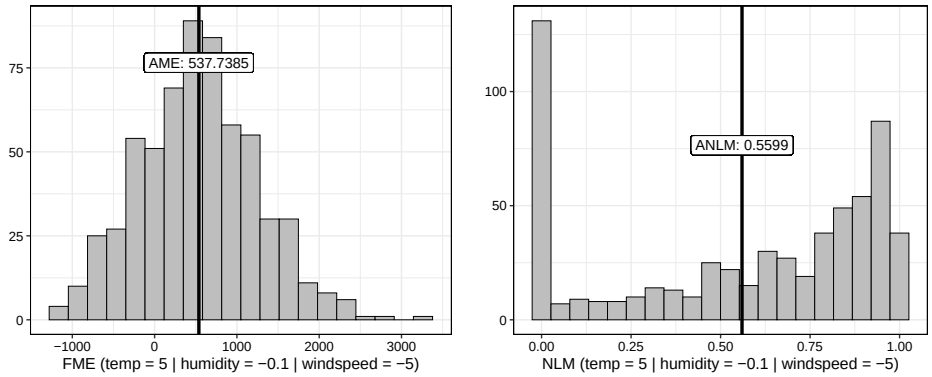


Figure 10: Adding a third feature change, a drop in windspeed by 5 miles per hour, and visualizing the distribution of FME and NLM values. For the NLM plot, negative NLMs are binned as 0. It follows that the ANLM value in the plot differs from the raw ANLM in the summary output.

So far, we have only evaluated changes in continuous features. In many applications, we are concerned with switching categories of categorical features, a way of counterfactual thinking inherent to the human thought process. Note that despite the allure of switching categories of categorical features, one needs to be aware of potential model extrapolations. To illustrate this, we switch each non-rainy day's precipitation status to rainfall. Rainfall has an average isolated effect of lowering daily rentals by 803 bikes (see Fig. 11).

```
> effects.categ = fme(
+   model = forest,
+   data = bikes,
+   features = list("weather" = "rain"))

> summary(effects.categ)

Forward Marginal Effects Object

Step type:
  categorical

Feature & reference category:
  weather, rain

Extrapolation point detection:
  none, EPs: 0 of 710 obs. (0 %)

Average Marginal Effect (AME):
  -802.8716

> plot(effects.categ)
```

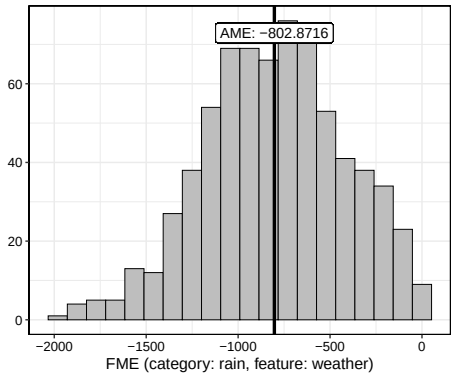


Figure 11: Distribution of categorical FMEs resulting from switching each non-rainy day's precipitation status to rain. On average, rainfall lowers predicted bike rentals by 803 bikes per day.

Regional explanations

In our examples, we can see highly heterogeneous local effects. The more heterogeneous FMEs are, the less information the AME carries. In many practical applications, we are interested in compactly describing the behavior of the predictive model across the feature space, akin to a beta coefficient in a linear model. This is where regional explanations come into play. We aim to find subgroups with more homogeneous FME values, thereby describing the behavior of the model not in terms of a global average but in terms of multiple regional averages (cAMEs).

In **fmeffects**, this can be achieved by further processing the `ForwardMarginalEffect` object containing FMEs (and optionally NLM values) using the `came()` function. This returns a `Partitioning` object (in this case, an object of the class `"PartitioningCTREE"`, a subclass of the abstract class `"Partitioning"`, see later section on Design and options for extensions).

For the univariate change in temperature by 5 degrees Celsius, we decide to search for precisely 2 subgroups[2] (for a description of this algorithm, see the following section on Design and options for extensions). A summary of the created object informs us about the number of observations, cAME, and standard deviation (SD) of FMEs inside the root node and leaf nodes (the found subgroups). We succeeded in finding subgroups with lower SDs while maintaining an appropriate sample size. The root node SD of 611 can be successfully split down to 437 and 355 within the subgroups. By visualizing the tree, we can see how the data was partitioned. For cooler outside temperatures equal to or lower than $\approx$ 16 degrees Celsius, we can observe a positive cAME of 728 additional bike rentals per day. On warmer days with a temperature above $\approx$ 16 degrees Celsius, the model predicts 196 less bike rentals a day when the outside temperature increases by 5 degrees.

```
> subspaces = came(effects = effects.univariate.temp, number.partitions = 2)
> summary(subspaces)

PartitioningCtree of an FME object

Method:  partitions = 2

   n      cAME  SD(fME)
 683  307.3275 611.0778 *
 372  728.3942 437.0463
 311 -196.3278 354.5090
---
* root node (non-partitioned)

AME (Global): 307.3275

> plot(subspaces)
```

[2]This value is to be set by the user depending on how many regional explanations are to be found. Alternatively, we can search for a pre-defined SD of FMEs inside the terminal nodes. How many subgroups can be found depends on the data and predictive model.

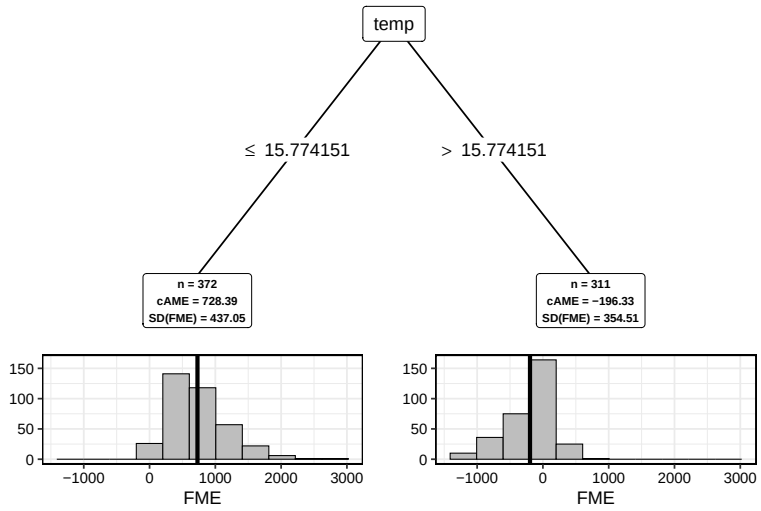


Figure 12: Using a decision tree to find subgroups of observations with more homogeneous FMEs of increasing 'temp' by 5 degrees Celsius. Each leaf node visualizes one subgroup, the number of observations, the cAME, and the SD of FMEs indicating FME homogeneity.

Global explanations

When to search for regional explanations thus depends on the heterogeneity of local effects. The `ame()` function provides an appropriate summary for the entire model. It uses a default step size of 1 or 0.01 for small feature ranges. For categorical FMEs, it uses every observed category as a reference category. Alternatively, custom step sizes and subsets of features can be used. The `summary()` function prints a compact model summary of each feature, a default step size, the AME, the SD of FMEs, 25% and 75% quantiles of FMEs, as well as the number of observations left after excluding extrapolation points (EPs). A large dispersion indicates heterogeneity of FMEs and thus a small fidelity of the AME and possible benefits from searching for subgroups with varying cAMEs. A different workflow can, therefore, also consist of starting with the table generated by `ame()` and deciding which feature effects can be described by AMEs and which might be better describable by subgroups and cAMEs. If this has been unsuccessful, we can resort to local model explanations. Recall our example from the previous section on Regional explanations where we split FMEs associated with increasing temperature by 5 degrees Celsius. From the `ame()` summary, we see that 'temp' has a relatively large SD in relation to its AME (here calculated with a step size of 1), and the interquartile range indicates a wide spread of FMEs from -20 in the 25% quantile up to 108 in the 75% quantile, which makes it a promising candidate to find subgroups with more homogeneous FMEs.

```
> ame.results = ame(model = forest, data = bikes)
> summary(ame.results)

Model Summary Using Average Marginal Effects:

      Feature step.size        AME       SD       0.25       0.75   n
1      season    winter  -942.0906 466.3691 -1298.1011  -617.5663 550
2      season    spring   136.2185 569.5307  -244.4237   650.0125 547
3      season    summer   293.6264 549.2972   -42.7551   738.2056 543
4      season      fall   533.5502 579.5541    52.3706  1138.0863 553
5        year         0 -1899.4966 639.1695 -2354.1389 -1506.0582 366
6        year         1  1790.6269 524.4711  1421.7925  2194.1396 365
7     holiday        no     195.93  218.386   123.2468   228.0909  21
8     holiday       yes  -133.3134 154.8869  -201.3635   -25.1245 710
9     weekday    Sunday   155.5219 188.8708     9.3486   252.0308 626
10    weekday    Monday  -158.9218 215.5047  -263.2441    -4.8485 626
11    weekday   Tuesday  -115.7316 193.4508  -197.7396    13.3208 626
```

```
12    weekday Wednesday   -44.3056 173.8664  -115.5562    63.1344 627
13    weekday  Thursday     16.005  161.125   -61.1673    89.5043 627
14    weekday    Friday    57.1498 163.5602   -27.6519    128.752 627
15    weekday  Saturday   103.7648 170.5678    -0.2044    178.493 627
16 workingday        no   -42.8794 139.8572  -145.7104    66.2131 500
17 workingday       yes    48.1298 158.3666   -60.2448   145.5003 231
18    weather     misty  -221.5664 328.3458  -413.4363   -69.4238 484
19    weather     clear   385.8674 347.6119   162.2048   476.8631 268
20    weather      rain  -802.8716 384.2624 -1054.7158  -543.2614 710
21       temp         1    58.0487 164.8714   -20.0019   108.4669 731
22   humidity      0.01     -19.86  62.1753   -36.5407    10.4535 731
23  windspeed         1   -24.7315  77.1757   -56.9247    13.7468 731
```

Design and options for extensions

The **fmeffects** package is built on a modular design for improved maintainability and future extensions. Fig. 13 provides a visual overview of the core design. The greatest emphasis is placed on the strategy and adapter design patterns (Gamma et al., 1994). Simply put, the strategy pattern decouples the source code for algorithm selection at runtime into separate classes. We repeatedly implement this pattern throughout the package by creating abstract classes whose subclasses implement specific functionalities. The adapter design pattern (also called a "wrapper") creates an interface for communication between two classes.

- `"Predictor"`: An abstract class that implements the adapter pattern to accommodate future implementations of storing a predictive model. `"PredictorMLR3"`, `"PredictorParsnip"`, and `"PredictorCaret"` are subclasses that store an **mlr3**, **parsnip** (Kuhn and Vaughan, 2023) (part of **tidymodels**), or **caret** model object. This allows users of **fmeffects** to use numerous predictive models such as random forests, gradient boosting, support vector machines, or neural networks. `"PredictorLM"` stores models returned by `lm()`,`glm()`,`or gam()`. The package can be extended with novel model types by implementing a new subclass that stores the model, data, target, and is able to return predictions.
- `"AverageMarginalEffects"`: A class to compute AMEs for each feature in the data (or a subset of features). Internally, a new `"ForwardMarginalEffect"` object is used to compute and aggregate FMEs. For convenience, we implement a wrapper function `ame()` to facilitate object creation and to initiate computations without requiring user input in the form of **R6** syntax.
- `"ForwardMarginalEffect"`: The centerpiece class of the package. It keeps access to a `Predictor`, stores important information to create FMEs, and after the computations are completed, stores results and gives access to visualization methods. For convenience, the wrapper function `fme()` can be used.
- `"FMEPlot"`: An abstract class for code decoupling of different plot categories into distinct classes. Subclasses include `"FMEPlotUnivariate"`, `"FMEPlotBivariate"`, `"FMEPlotHigherOrder"`, `"FMEPlotCategorical"`.
- `"ExtrapolationDetector"`: Identifies (and excludes) EPs. The current implementation supports the method "envelope", excluding points outside the multivariate envelope of the training data.
- `"NonLinearityMeasure"`: For the NLM, we need to approximate three line integrals, e.g., via Simpson's 3/8 rule. The general definition of Simpson's 3/8 rule for a univariate function f(x) and integration interval $[a, b]$ corresponds to:

$$\int_a^b f(x) \approx \frac{b-a}{8}\left[f(a) + 3f\left(\frac{2a+b}{3}\right) + 3f\left(\frac{a+2b}{3}\right) + f(b)\right] \tag{3}$$

  We make use of a composite Simpson rule, which divides up the interval $[a, b]$ into n subintervals of equal size and approximates each subinterval with Eq. (3).
- `"Partitioning"`: An abstract class, allowing for various implementations of finding subgroups for cAMEs. For convenience, the wrapper function `came()` can be used. The current implementation supports RP via the **rpart** and **partykit** (CTREE algorithm) packages (classes `"PartitioningRPart"` and `"PartitioningCTREE"`).

  We believe there are two criteria that should guide this process: FME homogeneity within each subgroup and the number of subgroups. A low number of subgroups is generally preferred. In certain applications, we may want to search for a predefined number of subgroups, akin to the search for a predefined number of clusters in clustering problems. Many RP algorithms do not support searching for a number of subgroups, which is what the `"Pruner"` class is intended for.

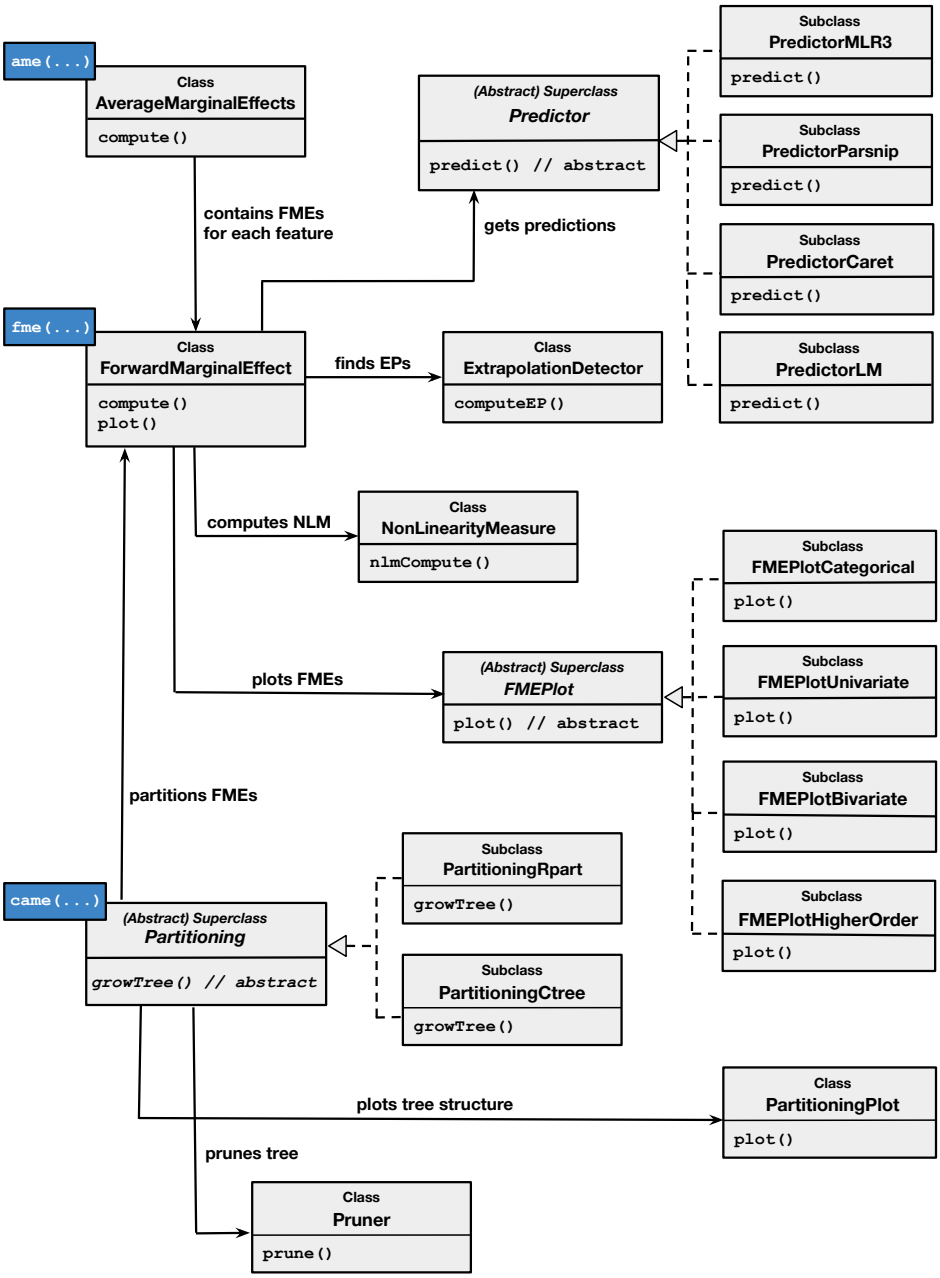


Figure 13: Design overview of the **fmeffects** package, including methods that implement the main functionality of each class. Classes may contain more methods than depicted. Blue boxes indicate wrapper functions to instantiate objects of the respective class.

- `"Pruner"`: To receive a predefined number of subgroups for arbitrary RP algorithms, we follow a two-stage process: grow a large tree by tweaking tree-specific hyperparameters and then prune it back to receive the desired number of subgroups. A `"Partitioning"` subclass is implemented such that it can first grow a large tree, e.g., with a low complexity parameter for **rpart**. Then `"Pruner"` iteratively computes the SD of FMEs for each parent node of the current terminal nodes and removes all terminal nodes of the parent with the lowest SD.
- `"PartitioningPlot"`: Decouples visualizations of the separation of $\mathcal{D}$ into subgroups from specific implementations of the `"Partitioning"` subclass. Here, we make use of a dependency on **partykit** for a tree data structure. This allows visualizations of any partitioning with the same methods. The package **ggparty** (Borkovec and Madin, 2019) creates tree figures that illustrate the partitioning, descriptive statistics for each terminal node, and histograms of FMEs (and optionally NLM values).

Conclusion

This paper introduces the R package **fmeffects**, the first software implementation of the theory surrounding FMEs. We showcase the package functionality with an applied use case and discuss design choices and implications for future extensions. FMEs are a versatile model-agnostic interpretation method and give us comprehensible model explanations in the form of: if we change x by an amount h, what is the change in predicted outcome $\widehat{y}$? FMEs equip stakeholders, including those without ML expertise, with the ability to understand feature effects for any model. We therefore hope that this package will work towards a more widespread adoption of FMEs in practice.

Software development is an ongoing process. As the theory surrounding FMEs evolves, so should the **fmeffects** package. As noted by Scholbeck et al. (2024), possible directions for future research include the development of techniques to better quantify extrapolation risk for the selection of step sizes; furthermore, the subgroup search for cAMEs is subject to uncertainties, which may be able to be quantified; and lastly, we may be able to spare computations by searching for representative FMEs, similar to prototype observations that are representative of clusters of observations (Tan et al., 2019). Future performance improvements may also be made via parallel computing, which at this point is only implemented for NLM computations.

Bibliography

A. Adadi and M. Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access*, 6:52138–52160, 2018. URL https://doi.org/10.1109/access.2018.2870052. [p7]

D. W. Apley and J. 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. URL https://doi.org/10.1111/rssb.12377. [p5, 7]

V. Arel-Bundock. *marginaleffects: Predictions, Comparisons, Slopes, Marginal Means, and Hypothesis Tests*, 2023. URL https://CRAN.R-project.org/package=marginaleffects. R package version 0.11.1. [p5]

S. Athey and G. W. Imbens. Machine learning methods that economists should know about. *Annual Review of Economics*, 11(1):685–725, 2019. URL https://doi.org/10.1146/annurev-economics-080217-053433. [p1]

T. Bartus. Estimation of marginal effects using margeff. *The Stata Journal*, 5(3):309 – 329, 2005. [p1, 5]

M. Borkovec and N. Madin. *ggparty: 'ggplot' Visualizations for the 'partykit' Package*, 2019. URL https://CRAN.R-project.org/package=ggparty. R package version 1.0.0. [p19]

A.-L. Boulesteix, M. N. Wright, S. Hoffmann, and I. R. König. Statistical learning approaches in the genetic epidemiology of complex diseases. *Human Genetics*, 139(1):73–84, 2020. URL https://doi.org/10.1007/s00439-019-01996-9. [p1]

L. Breiman. Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical Science*, 16(3):199 – 231, 2001. URL https://doi.org/10.1214/ss/1009213726. [p1, 6]

M. Britton. Vine: Visualizing statistical interactions in black box models. arXiv, 2019. URL https://doi.org/10.48550/arXiv.1904.00561. [p5, 6]

G. Casalicchio, C. Molnar, and B. Bischl. Visualizing the feature importance for black box models. In M. Berlingerio, F. Bonchi, T. Gärtner, N. Hurley, and G. Ifrim, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 655–670. Springer International Publishing, Cham, 2019. URL https://doi.org/10.1007/978-3-030-10925-7_40. [p5]

W. Chang. *R6: Encapsulated Classes with Reference Semantics*, 2021. URL https://CRAN.R-project.org/package=R6. R package version 2.5.1. [p7]

I. C. Covert, S. Lundberg, and S.-I. Lee. Understanding global feature contributions with additive importance measures. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. [p5]

P. D. Dueben and P. Bauer. Challenges and design choices for global weather and climate models based on machine learning. *Geoscientific Model Development*, 11(10):3999–4009, 2018. URL https://doi.org/10.5194/gmd-11-3999-2018. [p1]

D. B. Dwyer, P. Falkai, and N. Koutsouleris. Machine learning approaches for clinical psychology and psychiatry. *Annual Review of Clinical Psychology*, 14(1):91–118, 2018. URL https://doi.org/10.1146/annurev-clinpsy-032816-045037. [p1]

H. Fanaee-T. Bike Sharing Dataset. UCI Machine Learning Repository, 2013. URL https://doi.org/10.24432/C5W894. [p7]

J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Ann. Statist.*, 29(5): 1189–1232, 2001. URL https://doi.org/10.1214/aos/1013203451. [p5]

E. Gamma, R. Helm, R. Johnson, and J. M. Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional, 1st edition, 1994. [p17]

A. Goldstein, A. Kapelner, J. Bleich, and E. 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. URL https://doi.org/10.1080/10618600.2014.907095. [p5]

W. Greene. *Econometric Analysis*. Pearson International, 8th edition, 2019. [p1, 5]

J. Herbinger, B. Bischl, and G. Casalicchio. Repid: Regional effect plots with implicit interaction detection. In G. Camps-Valls, F. J. R. Ruiz, and I. Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 10209–10233. PMLR, 2022. [p5, 6]

G. Hooker. Diagnosing extrapolation: Tree-based density estimation. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 569–574, New York, NY, USA, 2004a. Association for Computing Machinery. [p7]

G. Hooker. Discovering additive structure in black box functions. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 575–580, New York, NY, USA, 2004b. ACM. URL http://doi.acm.org/10.1145/1014052.1014122. [p7]

G. Hooker. Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables. *Journal of Computational and Graphical Statistics*, 16(3):709–732, 2007. URL https://doi.org/10.1198/106186007X237892. [p7]

G. Hooker, L. Mentch, and S. Zhou. Unrestricted permutation forces extrapolation: Variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, 31 (6):82, 2021. URL https://doi.org/10.1007/s11222-021-10057-z. [p7]

T. Hothorn and A. Zeileis. partykit: A modular toolkit for recursive partytioning in R. *Journal of Machine Learning Research*, 16(118):3905–3909, 2015. [p4]

U. Kamath and J. Liu. Introduction to interpretability and explainability. In *Explainable Artificial Intelligence: An Introduction to Interpretable Machine Learning*, pages 1–26. Springer International Publishing, Cham, 2021. URL https://doi.org/10.1007/978-3-030-83356-5_1. [p1]

M. Kuhn and D. Vaughan. *parsnip: A Common API to Modeling and Analysis Functions*, 2023. URL https://CRAN.R-project.org/package=parsnip. R package version 1.1.1. [p17]

T. J. Leeper. *margins: Marginal effects for model objects*, 2018. URL https://CRAN.R-project.org/package=margins. R package version 0.3.23. [p5]

S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. [p5]

D. Lüdecke. ggeffects: Tidy data frames of marginal effects from regression models. *Journal of Open Source Software*, 3(26):772, 2018. URL https://doi.org/10.21105/joss.00772. [p5]

C. J. McCabe, M. A. Halvorson, K. M. King, X. Cao, and D. S. Kim. Interpreting interaction effects in generalized linear models of nonlinear probabilities and counts. *Multivariate Behavioral Research*, 57 (2-3):243–263, 2022. URL https://doi.org/10.1080/00273171.2020.1868966. [p1]

N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A survey on bias and fairness in machine learning. *ACM Comput. Surv.*, 54(6), 2021. URL https://doi.org/10.1145/3457607. [p4]

T. D. Mize, L. Doan, and J. S. Long. A general framework for comparing predictions and marginal effects across models. *Sociological Methodology*, 49(1):152–189, 2019. URL https://doi.org/10.1177/0081175019852763. [p5]

C. Molnar. *Interpretable Machine Learning*. 2nd edition, 2022. URL https://christophm.github.io/interpretable-ml-book. [p1, 9]

C. Molnar, B. Bischl, and G. Casalicchio. iml: An R package for interpretable machine learning. *JOSS*, 3(26):786, 2018. URL https://doi.org/10.21105/joss.00786. [p9]

C. Molnar, G. König, J. Herbinger, T. Freiesleben, S. Dandl, C. A. Scholbeck, G. Casalicchio, M. Grosse-Wentrup, and B. Bischl. General pitfalls of model-agnostic interpretation methods for machine learning models. In A. Holzinger, R. Goebel, R. Fong, T. Moon, K.-R. Müller, and W. Samek, editors, *xxAI - Beyond Explainable AI. xxAI 2020. Lecture Notes in Computer Science, vol 13200*, Cham, 2022. Springer. URL https://doi.org/10.1007/978-3-031-04083-2_4. [p7]

C. Molnar, G. König, B. Bischl, and G. Casalicchio. Model-agnostic feature importance and effects with dependent features: A conditional subgroup approach. *Data Mining and Knowledge Discovery*, 38(5): 2903–2941, 2024. URL https://doi.org/10.1007/s10618-022-00901-9. [p5]

S. Mullainathan and J. Spiess. Machine learning: An applied econometric approach. *Journal of Economic Perspectives*, 31(2):87–106, 2017. URL https://doi.org/10.1257/jep.31.2.87. [p1]

E. Onukwugha, J. Bergtold, and R. Jain. A primer on marginal effects—part I: Theory and formulae. *PharmacoEconomics*, 33(1):25–30, 2015. URL https://doi.org/10.1007/s40273-014-0210-6. [p1]

A. Rajkomar, J. Dean, and I. Kohane. Machine learning in medicine. *New England Journal of Medicine*, 380(14):1347–1358, 2019. URL https://doi.org/10.1056/NEJMra1814259. [p1]

M. T. Ribeiro, S. Singh, and C. 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*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. URL https://doi.org/10.1145/2939672.2939778. [p5]

C. A. Scholbeck, C. Molnar, C. Heumann, B. Bischl, and G. Casalicchio. Sampling, intervention, prediction, aggregation: A generalized framework for model-agnostic interpretations. In P. Cellier and K. Driessens, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 1167 of *Communications in Computer and Information Science*, pages 205–216. Springer International Publishing, Cham, 2020. URL https://doi.org/10.1007/978-3-030-43823-4_18. [p1, 5]

C. A. Scholbeck, J. Moosbauer, G. Casalicchio, H. Gupta, B. Bischl, and C. Heumann. Position paper: Bridging the gap between machine learning and sensitivity analysis. arXiv, 2023. URL https://doi.org/10.48550/arXiv.2312.13234. [p5]

C. A. Scholbeck, G. Casalicchio, C. Molnar, B. Bischl, and C. Heumann. Marginal effects for non-linear prediction functions. *Data Mining and Knowledge Discovery*, 38(5):2997–3042, 2024. URL https://doi.org/10.1007/s10618-023-00993-x. [p1, 2, 3, 4, 5, 7, 19]

StataCorp. *Stata: Release 18*. College Station, TX: StataCorp LLC., 2023. [p5]

E. Štrumbelj and I. Kononenko. An efficient explanation of individual classifications using game theory. *Journal of Machine Learning Research*, 11(1):1–18, 2010. [p5]

P.-N. Tan, A. Karpatne, M. Steinbach, and V. Kumar. *Introduction to Data Mining: Global Edition*. Pearson, 2019. [p19]

T. Therneau and B. Atkinson. *rpart: Recursive Partitioning and Regression Trees*, 2019. URL https://CRAN.R-project.org/package=rpart. R package version 4.1-15. [p4]

S. Wachter, B. Mittelstadt, and C. Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harvard Journal of Law and Technology*, 31(2):841–887, 2018. [p5]

R. Williams. Using the margins command to estimate and interpret adjusted predictions and marginal effects. *Stata Journal*, 12(2):308–331(24), 2012. [p1, 2, 5]

M. N. Wright and A. Ziegler. ranger: A fast implementation of random forests for high dimensional data in C++ and R. *Journal of Statistical Software*, 77(1), 2017. URL http://dx.doi.org/10.18637/jss.v077.i01. [p7]

Holger Löwe
Ludwig-Maximilians-Universität in Munich
Germany
hbj.loewe@gmail.com

Christian A. Scholbeck
Ludwig-Maximilians-Universität in Munich
Munich Center for Machine Learning (MCML)
Germany
https://orcid.org/0000-0001-6607-4895
christian.scholbeck@stat.uni-muenchen.de

Christian Heumann
Ludwig-Maximilians-Universität in Munich
Germany
christian.heumann@stat.uni-muenchen.de

Bernd Bischl
Ludwig-Maximilians-Universität in Munich
Munich Center for Machine Learning (MCML)
Germany
bernd.bischl@stat.uni-muenchen.de

Giuseppe Casalicchio
Ludwig-Maximilians-Universität in Munich
Munich Center for Machine Learning (MCML)
Germany
giuseppe.casalicchio@stat.uni-muenchen.de

10 | Algorithm-Agnostic Feature Attributions for Clustering

Contributing Paper

Scholbeck, C. A., Funk, H., and Casalicchio, G. (2023a). "Algorithm-Agnostic Feature Attributions for Clustering". In: *Explainable Artificial Intelligence: First World Conference, xAI 2023, Lisbon, Portugal, July 26–28, 2023, Proceedings, Part I.* Ed. by Longo, L. Vol. 1901. Communications in Computer and Information Science. Cham: Springer Nature Switzerland, pp. 217–240. DOI: 10.1007/978-3-031-44064-9_13

Declaration of Contributions

C.A. Scholbeck and H. Funk share the first authorship of this paper. C.A. Scholbeck and G. Casalicchio developed the initial project idea with equal contributions. The paper builds upon the master thesis of H. Funk[1], which was supervised by C.A. Scholbeck and G. Casalicchio under close academic guidance and supervision. The paper was drafted by C.A. Scholbeck based on the material in H. Funk's master thesis with different terminology, notation, and presentation. The paper extends the master thesis in multiple directions: it introduces the novel FACT framework of work stages for cluster explanation methods (sampling, intervention, reassignment, aggregation), an additional simulation (Section 5.4), and a proof on the equivalency between SMART with a micro-averaged F1 score and G2PC (Theorem 1). C.A. Scholbeck led the submission of the final edition and revised the paper according to the feedback from his co-authors and external reviewers.

H. Funk and G. Casalicchio contributed to the formulation of the project's research objectives and assisted in revising the paper. H. Funk developed the idea for SMART; implemented all methods with continuous support from G. Casalicchio and C.A. Scholbeck; created all simulations, the applied example, and the R package `FACT`; and suggested several notable modifications. G. Casalicchio coordinated the project, provided valuable support, and suggested several notable modifications.

[1] Funk, Henri (2022): Towards Algorithm-Agnostic Interpretability in Clustering. Master Thesis, Ludwig-Maximilians-Universität München

Algorithm-Agnostic Feature Attributions for Clustering

Christian A. Scholbeck[1,2(✉)], Henri Funk[1,2], and Giuseppe Casalicchio[1,2]

[1] LMU Munich, Munich, Germany
[2] Munich Center for Machine Learning (MCML), Munich, Germany
{christian.scholbeck,henri.funk,
giuseppe.casalicchio}@stat.uni-muenchen.de

Abstract. Understanding how assignments of instances to clusters can be attributed to the features can be vital in many applications. However, research to provide such feature attributions has been limited. Clustering algorithms with built-in explanations are scarce. Common algorithm-agnostic approaches involve dimension reduction and subsequent visualization, which transforms the original features used to cluster the data; or training a supervised learning classifier on the found cluster labels, which adds additional and intractable complexity. We present FACT (feature attributions for clustering), an algorithm-agnostic framework that preserves the integrity of the data and does not introduce additional models. As the defining characteristic of FACT, we introduce a set of work stages: sampling, intervention, reassignment, and aggregation. Furthermore, we propose two novel FACT methods: SMART (scoring metric after permutation) measures changes in cluster assignments by custom scoring functions after permuting selected features; IDEA (isolated effect on assignment) indicates local and global changes in cluster assignments after making uniform changes to selected features.

Keywords: Interpretable clustering · explainable AI · feature attributions · algorithm-agnostic · effect · importance · FACT · SMART · IDEA

1 Introduction

Recent efforts have focused on making machine learning models interpretable, both via model-agnostic interpretation methods and novel interpretable model types [27], which is referred to as interpretable machine learning or explainable artificial intelligence in different contexts. Unfortunately, success in addressing cluster interpretability has been limited [3]. In the context of our paper, feature attributions (FAs) either provide information regarding the importance of features for assigning instances to clusters (overall and to specific clusters); or how isolated changes in feature values affect the assignment of single instances or

C. A. Scholbeck and H. Funk—Contributed equally.

L. Longo (Ed.): xAI 2023, CCIS 1901, pp. 217–240, 2023.
https://doi.org/10.1007/978-3-031-44064-9_13

the entire data set to each cluster. Interpretable clustering algorithms [3,23,31] provide some insight into the constitution of clusters, e.g., relationships between features within clusters, but often fall short of providing FAs. Furthermore, the range of interpretable clustering algorithms is limited. An alternative approach is to post-process the original data (e.g., via principal components analysis) and visualize the found clusters in a lower-dimensional space [17]. This obfuscates interpretations by transforming the original features used to cluster the data. A third option is to train a supervised learning (SL) classifier on the found cluster labels, which is interpreted instead. This adds additional and intractable complexity on top of the clustering by introducing an additional model.

Contributions: We present FACT[1] (feature attributions for clustering), a framework that is compatible with any clustering algorithm able to reassign instances to clusters (algorithm-agnostic), preserves the integrity of the data, and does not introduce additional models. As the defining characteristic of FACT, we propose four work stages: sampling, intervention, reassignment, and aggregation. Furthermore, we introduce two novel FACT methods: SMART (scoring metric after permutation) measures changes in cluster assignments by custom scoring functions after permuting selected features; IDEA (isolated effect on assignment) indicates local and global changes in cluster assignments after making uniform changes to selected features. FACT is inspired by principles of model-agnostic interpretation methods in SL, which detach the interpretation method from the model, thereby detaching the interpretation method from the clustering algorithm. In Fig. 1, we summarize how SMART and IDEA utilize select ideas from SL and how they innovate with new principles.

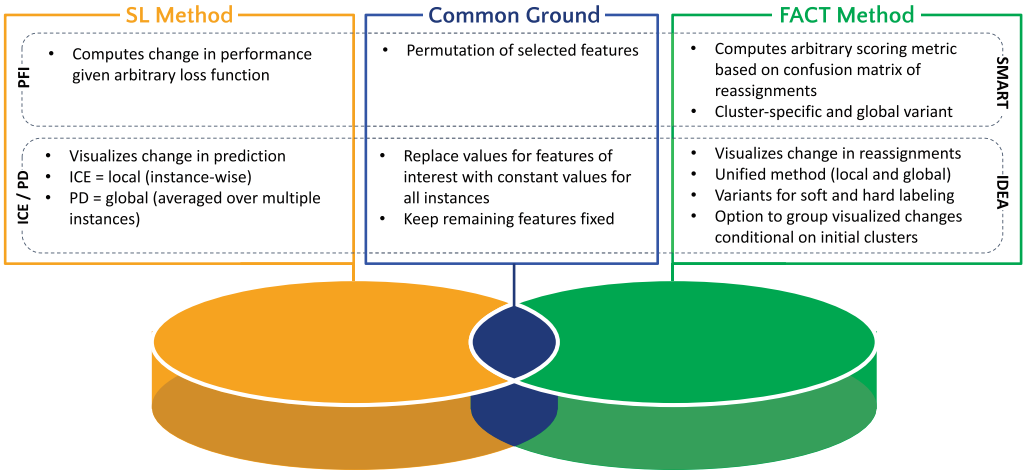


Fig. 1. Comparison of related concepts from SL (overlap in the center) with the clustering setting and novelties for FACT methods SMART and IDEA (right side).

[1] All presented methods are implemented in the R package `FACT` [13].

2 Notation and Preliminaries

2.1 Notation

We cluster a data set $\mathcal{D} = \left\{\mathbf{x}^{(i)}\right\}_{i=1}^{n}$ (where $\mathbf{x}^{(i)}$ denotes the i-th observation) into k clusters $\mathcal{D}^{(c)},\ c \in \{1, \ldots, k\}$. A single observation $\mathbf{x}$ consists of p feature values $\mathbf{x} = (x_1, \ldots, x_p)$. A subset of features is denoted by $S \subseteq \{1, \ldots, p\}$ with the complement set being denoted by $-S = \{1, \ldots, p\} \setminus S$. With slight abuse of notation, an observation $\mathbf{x}$ can be partitioned into $\mathbf{x} = (\mathbf{x}_S, \mathbf{x}_{-S})$, regardless of the order of elements within $\mathbf{x}_S$ and $\mathbf{x}_{-S}$. A data set $\mathcal{D}$ where all features in S have been shuffled jointly is denoted by $\tilde{\mathcal{D}}_S$. The initial clustering is encoded within a function f that - conditional on whether the clustering algorithm outputs hard or soft labels[2] - maps each observation $\mathbf{x}$ to a cluster c (hard label) or to k soft labels:

$$\text{Hard labeling: } f : \mathbf{x} \mapsto c,\ c \in \{1, \ldots, k\}$$
$$\text{Soft labeling: } f : \mathbf{x} \mapsto \mathbb{R}^k$$

For soft clustering algorithms, $f^{(c)}(\mathbf{x})$ denotes the soft label for the c-th cluster. This notation is also used to indicate the cluster-specific value within an IDEA vector (see Sect. 3.2).

2.2 Interpretations of Supervised Learning Models

In recent years, the interpretation of model output has become a popular research topic [28]. Existing techniques provide explanations in terms of FAs (e.g., a value indicating a feature's importance to the model or a curve indicating its effects on the prediction), model internals (e.g., beta coefficients for linear regression models), data points (e.g., counterfactual explanations [39]), or surrogate models (i.e., interpretable approximations to the original model) [27]. Many model-agnostic methods are based on identical work stages: First, a subset of observations is sampled which we intend to use for the model interpretation (sampling stage). This is followed by an intervention in feature values where the instances from the sampling stage are manipulated in certain ways (intervention stage). Next, we predict with the trained model and this new, artificial data set (prediction stage). This produces local (observation-wise) interpretations which can be further aggregated to produce global or semi-global interpretations (aggregation stage) [35]. These work stages can be considered a sensitivity analysis (SA) of the model.

[2] A vector of soft labels represents the propensity of an observation being assigned to each cluster. A convenient representation corresponds to a vector of pseudo probabilities $[0, 1]^k$. We refrain from labeling any algorithm as a hard or soft clustering algorithm because often an algorithm can output both hard and soft labels, e.g., k-means - traditionally considered a hard clustering algorithm - could output soft labels in the form of Euclidean distances to each cluster centroid.

Established methods to determine FAs for SL models comprise the individual conditional expectation (ICE) [16], partial dependence (PD) [11], accumulated local effects (ALE) [2], local interpretable model-agnostic explanations (LIME) [33], Shapley values [26,37], or the permutation feature importance (PFI) [6, 9]. The functional analysis of variance (FANOVA) [18,34] and Sobol indices [36] of a high-dimensional model representation are powerful tools to quantify input influence on the model output in terms of variance but are limited by the requirement for independent inputs. Among the mentioned techniques, the following three are useful for the development of SMART and IDEA:

- **PFI:** Shuffling a feature in the data set destroys the information it contains. The PFI evaluates the model performance before and after shuffling and uses the change in performance to describe a feature's importance.
- **ICE:** The ICE function indicates the prediction of an SL model for a single observation $\mathbf{x}$ where a subset of values $\mathbf{x}_S$ is replaced with values $\tilde{\mathbf{x}}_S$ while we condition on the remaining features $\mathbf{x}_{-S}$, i.e., keep them fixed. For single features of interest, an ICE corresponds to a single curve.
- **PD:** The PD function indicates the expected prediction given the marginal effect of a set of features. The PD can be estimated through a point-wise aggregation of ICEs across all considered instances.

2.3 Interpretations for Clustering Algorithms

Unsupervised clustering has largely been ignored by this line of research. However, for high-dimensional data sets, the clustering routine can often be considered a black box, as we may not be able to assess and visualize the multidimensional cluster patterns found by the algorithm. It is, therefore, desirable to receive deeper explanations of how an algorithm's decisions can be attributed to the features. Interpretable clustering algorithms incorporate the interpretability criterion directly into the cluster search. One option is to find an interpretable tree-based clustering [5,10,12,14,15,24,25,30]. Interpretable clustering of numerical and categorical objects (INCONCO) [31] is an information-theoretic approach based on finding clusters that minimize minimum description length. It finds simple rule descriptions of the clusters by assuming a multivariate normal distribution and taking advantage of its mathematical properties. Interpretable clustering via optimal trees (ICOT) [3] uses decision trees to optimize a cluster quality measure. In [23] clusters are explained by forming polytopes around them. Mixed integer optimization is used to jointly find clusters and define polytopes.

The focus of this paper lies on algorithm-agnostic interpretations. In many cases, we wish to use a clustering algorithm that does not provide any explanations. Furthermore, even interpretable clustering algorithms often do not directly provide FAs, thus still requiring additional interpretation methods. Analogously to SL, we may define post-hoc interpretations (which are typically algorithm-agnostic) as ones that are obtained after the clustering procedure, e.g., by showing a subset of representative elements of a cluster or via visualization techniques

such as scatter plots [22]. In most cases, the data is high-dimensional and requires the use of dimensionality reduction techniques such as principal component analysis (PCA) before being visualized in two or three dimensions. PCA creates linear combinations of the original features called the principal components (PCs). The goal is to select fewer PCs than original features while still explaining most of their variance. PCA obscures the information contained in the original features by rotating the system of coordinates. For instance, interpretable correlation clustering (ICC) [1] uses post-processing of correlation clusters. A correlation cluster groups the data such that there is a common within-cluster hyperplane of arbitrary dimensionality. ICC applies PCA to each correlation cluster's covariance matrix, thereby revealing linear patterns inside the cluster. One can also use an SL algorithm to post-process the clustering outcome which learns to find interpretable patterns between the found cluster labels and the features. Although we may use any SL algorithm, classification trees are a suitable choice due to naturally providing decision rules on how they arrive at a prediction [4]. Although this is a simple approach that can produce FAs via model internals or model-agnostic interpretation methods, it introduces intractable complexity through an additional model.

An algorithm-agnostic option that bypasses these issues is a form of SA where data are deliberately manipulated and reassigned to existing clusters. The global permutation percent change (G2PC) [8] indicates the percentage of change between the cluster assignments of the original data and those from a permuted data set. A high G2PC indicates an important feature for the clustering outcome. The local permutation percent change (L2PC) [8] uses the same principle for single instances.

3 FACT Framework and Methods

We first define a distinction of various FAs for the clustering setting: A *local FA* indicates how a feature contributes to the cluster assignment of a single observation; a *global FA* indicates how a feature contributes to the cluster assignments of an entire data set; a *cluster-specific FA* indicates how a feature contributes to the assignments of observations to one specific cluster. We introduce four work stages for FACT methods:

- **Sampling:** We sample a subset of observations that were previously clustered and shall be used to determine FAs. The larger this subset, the better our FA estimates. The smaller, the faster their computation.
- **Intervention:** Next, we manipulate feature values for the subset of observations from the sampling stage. This can be a targeted intervention (e.g., replacing current values with a pre-defined value) or shuffling values.
- **Reassignment:** This new, manipulated data set is reassigned to existing clusters through soft or hard labels. For each observation from the sampling stage, we receive a vector of soft labels or a single hard label.

– **Aggregation:** The soft or hard labels from the reassignment stage are aggregated in various ways, e.g., they can be averaged (soft labels) or counted (hard labels) cluster-wise.

The only prerequisite is an existing clustering based on an algorithm that can reassign instances to existing clusters through soft or hard labels. Methods only differ with respect to the intervention and aggregation stages. Next, we present our two novel FACT methods SMART and IDEA.

3.1 Scoring Metric After Permutation (SMART)

The intervention stage consists of shuffling values for a subset of features S in the data set $\mathcal{D}$ (i.e., jointly shuffling rows for a subset of columns); the aggregation stage consists of measuring the change in cluster assignments through an appropriate scoring function h applied to a confusion matrix consisting of original cluster assignments and cluster assignments after shuffling. When comparing original cluster assignments and the ones after shuffling the data, we can create a confusion matrix (see Appendix A) in the same way as in multi-class classification. One option to evaluate the confusion matrix is to directly use a scoring metric suitable for multiple clusters, e.g., the percentage of observations changing clusters after the intervention as in G2PC (found in all non-diagonal elements of the confusion matrix, see Eq. (1) for a definition). If one is interested in a scoring metric specifically developed for binary confusion matrices, the alternative is to consider binary comparisons of cluster c versus the remaining clusters. The results of all binary comparisons can then be aggregated either through a micro or a macro-averaged score (see Appendix B). Established scoring metrics based on binary confusion matrices include the F1 score (see Appendix B), Rand [32], or Jaccard [21] index. The micro-averaged score (hereafter referred to as micro score) is a suitable metric if all instances shall be considered equally important. The macro-averaged score (hereafter referred to as macro score) suits a setting where all classes (i.e., clusters in our case) shall be considered equally important. In general terms, the scoring function maps a confusion matrix to a scalar scoring metric. A multi-cluster scoring function is defined as:

$$h_{\text{multi}} : \mathbb{N}_0^{k \times k} \mapsto \mathbb{R}$$

A binary scoring function is defined as:

$$h_{\text{binary}} : \mathbb{N}_0^{2 \times 2} \mapsto \mathbb{R}$$

Let $M \in \mathbb{N}_0^{k \times k}$ denote the multi-cluster confusion matrix and $M_c \in \mathbb{N}_0^{2 \times 2}$ the binary confusion matrix for cluster c versus the remaining clusters (see Appendix A for details). SMART for feature set S corresponds to:

$$\begin{aligned} \text{Multi-cluster scoring:} \quad & \text{SMART}(\mathcal{D}, \tilde{\mathcal{D}}_S) = h_{\text{multi}}(M) \\ \text{Binary scoring:} \quad & \text{SMART}(\mathcal{D}, \tilde{\mathcal{D}}_S) = \text{AVE}(h_{\text{binary}}(M_1), \ldots, h_{\text{binary}}(M_k)) \end{aligned}$$

where AVE averages a vector of binary scores, e.g., via micro or macro averaging. In order to reduce variance in the estimate from shuffling the data, one can shuffle t times and evaluate the distribution of scores. Let $\tilde{\mathcal{D}}_S^{(t)}$ denote the t-th shuffling iteration for feature set S. The SMART point estimate is given by:

$$\overline{\text{SMART}}(\mathcal{D}, \tilde{\mathcal{D}}_S) = \psi\left(\text{SMART}(\mathcal{D}, \tilde{\mathcal{D}}_S^{(1)}), \ldots, \text{SMART}(\mathcal{D}, \tilde{\mathcal{D}}_S^{(t)})\right)$$

where ψ extracts a sample statistic such as the mean or median.

We can demonstrate the equivalency between directly applying the G2PC scoring metric to the confusion matrix and micro averaging F1 scores[3]. Given a multi-cluster confusion matrix M (see Appendix A), G2PC is defined as:

$$\begin{aligned}\text{G2PC}(M) &= \frac{1}{n}\left(\sum_{i=1}^{k}\sum_{j=1}^{k} \#_{ij} - \sum_{l=1}^{k} \#_{ll}\right) \\ &= \frac{1}{n}\left(n - \sum_{l=1}^{k} \#_{ll}\right) \\ &= 1 - \frac{1}{n}\sum_{l=1}^{k} \#_{ll} \end{aligned} \tag{1}$$

The micro F1 score is equivalent to accuracy (for settings where each instance is assigned a single label), so the following relation holds (refer to Appendix D for a detailed proof):

Theorem 1 (Equivalency between SMART with micro F1 and G2PC).

$$1 - \text{G2PC}(M) = \text{AVE}_{\text{MICRO}}(\text{F1}(M_1), \ldots, \text{F1}(M_k)) = \text{F1}_{\text{micro}}(M)$$

*Proof sketch. In our utilization of confusion matrices, a "false classification" corresponds to a change in clusters after the intervention, and a "true classification" corresponds to an observation staying in the same cluster. It follows that accuracy (*ACC*) represents the global percentage of observations staying in the initial cluster after the intervention stage:* $1 - \text{ACC}(M) = \text{G2PC}(M)$.

$\text{AVE}_{\text{MICRO}}(\text{F1}(M_1), \ldots, \text{F1}(M_k))$ *can be directly derived from the multi-cluster matrix* M *and is denoted by* $\text{F1}_{\text{micro}}(M)$. *Let* TP *denote the number of true positive labels,* FP *the number of false positives, and* FN *the number of false negatives. For multi-class classification problems,* $\text{FP} = \text{FN}$ *and thus:*

$$\text{F1}_{\text{micro}}(M) = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \text{ACC}(M)$$

It follows that $1 - \text{G2PC}(M) = \text{F1}_{\text{micro}}(M)$. □

[3] Micro averaging refers to a strategy of aggregating binary comparisons where each instance is considered equally important. For the F1 score, the equivalency can be directly derived from the multi-cluster confusion matrix and involves summing up all diagonal elements (true positives) and remaining elements (false positives or false negatives). See Appendices B and D for details.

Micro F1 scores are unsuited for unbalanced classes in classification settings, as they treat each instance as equally important. From the direct dependency between G2PC and micro F1, it follows that for clusters that considerably differ in size (i.e., imbalanced clusters), G2PC does not accurately represent the importance of features, as it is dominated by larger clusters. SMART in turn allows more flexible interpretations than G2PC, e.g., by using macro F1 scores.

We can also directly evaluate binary comparisons of the found clusters to obtain cluster-specific FAs. Recall that a cluster-specific FA provides information regarding how a feature influences reassignments of instances to one specific cluster. Algorithms 1 and 2 describe the cluster-specific and global SMART algorithms, respectively. The algorithms are applied in Sects. 5 and 6. See Fig. 10 for visualized outcomes. Note that the resampling procedure to reduce the variance of estimates is optional and that global SMART can also involve binary comparisons (which requires running cluster-specific SMART), e.g., via macro averaging; we circumscribe all such different variants as the computation of the multi-cluster score h.

Algorithm 1. Cluster-Specific SMART

```
run clustering algorithm
for all iter ∈ {1, ..., t} do
    shuffle columns S
    compute hard labels
    for all c ∈ {1, ..., k} do
        create a binary confusion matrix
        compute score h_c^(iter) from confusion matrix
    end for
end for
for all c ∈ {1, ..., k} do
    evaluate distribution of {h_c^(iter)}_{iter∈{1,...,t}}
end for
```

Algorithm 2. Global SMART

```
run clustering algorithm
for all iter ∈ {1, ..., t} do
    shuffle columns S
    compute hard labels
    create a multi-cluster confusion matrix
    compute multi-cluster score h^(iter)
end for
evaluate distribution of {h^(iter)}_{iter∈{1,...,t}}
```

3.2 Isolated Effect on Assignment (IDEA)

IDEA for soft labeling algorithms (sIDEA) indicates the soft label that an observation $\mathbf{x}$ with replaced values $\tilde{\mathbf{x}}_S$ is assigned to each c-th cluster. IDEA for hard labeling algorithms (hIDEA) indicates the cluster assignment of an observation $\mathbf{x}$ with replaced values $\tilde{\mathbf{x}}_S$. Both are described by the clustering (assignment) function f:

$$\text{IDEA}_{\mathbf{x}}(\tilde{\mathbf{x}}_S) = \text{sIDEA}_{\mathbf{x}}(\tilde{\mathbf{x}}_S) = \text{hIDEA}_{\mathbf{x}}(\tilde{\mathbf{x}}_S) = f(\tilde{\mathbf{x}}_S, \mathbf{x}_{-S})$$

sIDEA corresponds to a k-way vector:

$$\begin{aligned}\text{sIDEA}_{\mathbf{x}}(\tilde{\mathbf{x}}_S) &= \left(f^{(1)}(\tilde{\mathbf{x}}_S, \mathbf{x}_{-S}), \ldots, f^{(k)}(\tilde{\mathbf{x}}_S, \mathbf{x}_{-S})\right) \\ &= \left(\text{sIDEA}^{(1)}_{\mathbf{x}}(\tilde{\mathbf{x}}_S), \ldots, \text{sIDEA}^{(k)}_{\mathbf{x}}(\tilde{\mathbf{x}}_S)\right)\end{aligned}$$

Note that although IDEA is a local method, we typically compute it for a subset of observations selected in the sampling stage. The intervention stage consists of replacing $\mathbf{x}_S$ (for an observation $\mathbf{x}$) by $\tilde{\mathbf{x}}_S$. Algorithm 3 describes the computation of the local IDEA.

Algorithm 3. Local IDEA

```
run clustering algorithm
sample m vectors of feature values {x̃_S^(j)}_{j∈{1,...,m}}
for all i ∈ {1,...,n} do
    for all j ∈ {1,...,m} do
        generate hypothetical observation x = (x̃_S^(j), x_{-S}^(i))
        IDEA_{x^(i)}(x̃_S^(j)) = f(x)
    end for
end for
```

During the aggregation stage, we aggregate local IDEAs to a global function. For soft labeling algorithms, we can compute a point-wise average of soft labels for each cluster; for hard labeling algorithms, we can compute the fraction of hard labels for each cluster. The global IDEA is denoted by the corresponding data set $\mathcal{D}$. The global sIDEA corresponds to:

$$\text{sIDEA}_{\mathcal{D}}(\tilde{\mathbf{x}}_S) = \left(\frac{1}{n}\sum_{i=1}^{n}\text{sIDEA}^{(1)}_{\mathbf{x}^{(i)}}(\tilde{\mathbf{x}}_S), \ldots, \frac{1}{n}\sum_{i=1}^{n}\text{sIDEA}^{(k)}_{\mathbf{x}^{(i)}}(\tilde{\mathbf{x}}_S)\right) \tag{2}$$

where the c-th vector element is the average c-th element of local sIDEA vectors. The global hIDEA corresponds to:

$$\text{hIDEA}_{\mathcal{D}}(\tilde{\mathbf{x}}_S) = \left(\frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_1(\text{hIDEA}_{\mathbf{x}^{(i)}}(\tilde{\mathbf{x}}_S)), \ldots, \frac{1}{n}\sum_{i=1}^{n}\mathbb{1}_k(\text{hIDEA}_{\mathbf{x}^{(i)}}(\tilde{\mathbf{x}}_S))\right) \tag{3}$$

where the c-th vector element is the fraction of hard label reassignments to the c-th cluster. Algorithm 4 describes the computation of the global IDEA. See Sects. 5 and 6 for applications of the local and global IDEA and Figs. 6, 7, and 11 for visualizations.

A useful interpretation for hard labeling algorithms can be obtained by visualizing the percentage of all labels per isolated intervention. The fraction of the most frequent hard label indicates the – as we call it – "certainty" of the global IDEA function for hard labeling algorithms (see Fig. 6 on the left).

Whether the global IDEA can serve as a good description of the feature effect on the reassignment depends on the heterogeneity of underlying local effects. If substituting a feature set by the same values for all instances results in similar reassignments for most instances, the global IDEA is a good interpretation instrument. Otherwise, further investigations into the underlying local effects are required.

Algorithm 4. Global IDEA

```
run clustering algorithm
sample m vectors of feature values {x̃_S^(j)}_{j∈{1,...,m}}
for all i ∈ {1, ..., n} do
    compute IDEA_{x^(i)} (see Algorithm 3)
end for
for j ∈ {1, ..., m} do
    for c ∈ {1, ..., k} do
        if soft labeling algorithm then
            compute sIDEA_D^(c)(x̃_S^(j)) (see Eq. 2)
        else
            compute hIDEA_D^(c)(x̃_S^(j)) (see Eq. 3)
        end if
    end for
end for
```

Initial Cluster Effect on IDEA: If there is a certain within-cluster homogeneity, we ought to see similar shapes of local IDEA functions depending on the observations' initial cluster (before the intervention stage). Let c_{init} denote the initial cluster index. We receive one aggregate IDEA per initial cluster (we refrain from using the word "global" here, as there is a separate, global IDEA independent from the initial cluster), which reflects the aggregate, isolated effect of an intervention in the feature(s) of interest on the assignment to cluster c **per initial cluster** c_{init}:

$$\text{IDEA}_{\mathcal{D}^{(c_{\text{init}})}}(\tilde{\mathbf{x}}_S) = \left(\text{IDEA}^{(1)}_{\mathcal{D}^{(c_{\text{init}})}}(\tilde{\mathbf{x}}_S), \ldots, \text{IDEA}^{(k)}_{\mathcal{D}^{(c_{\text{init}})}}(\tilde{\mathbf{x}}_S)\right) \tag{4}$$

whose components correspond to (depending on the clustering algorithm output):

$$\text{sIDEA}^{(c)}_{\mathcal{D}^{(c_{\text{init}})}}(\tilde{\mathbf{x}}_S) = \frac{1}{n^{(c_{\text{init}})}} \sum_{i\,:\,\mathbf{x}^{(i)} \in \mathcal{D}^{(c_{\text{init}})}} \text{sIDEA}^{(c)}_{\mathbf{x}^{(i)}}(\tilde{\mathbf{x}}_S)$$

$$\text{hIDEA}^{(c)}_{\mathcal{D}^{(c_{\text{init}})}}(\tilde{\mathbf{x}}_S) = \frac{1}{n^{(c_{\text{init}})}} \sum_{i\,:\,\mathbf{x}^{(i)} \in \mathcal{D}^{(c_{\text{init}})}} \mathbb{1}_c(\text{hIDEA}_{\mathbf{x}^{(i)}}(\tilde{\mathbf{x}}_S))$$

where $n^{(c_{\text{init}})}$ corresponds to the number of observations within initial cluster c_{init}. This definition lends itself to a convenient visualization per initial cluster, which we showcase in Fig. 7.

4 Additional Notes on FACT

How to Generate Feature Values for Interventions: A simple option is to use a feature's sample distribution, i.e., all observed values. In classical SA of model output [34], one typically intends to explore the feature space as thoroughly as possible (space-filling designs). In SL, there are valid arguments against space-filling designs due to potential model extrapolations, i.e., predictions in areas where the model was not trained with enough data [19,29]. In clustering, the absence of model performance issues allows us to fill the feature space as extensively as possible, e.g., with unit distributions, random, or quasi-random (also referred to as low-discrepancy) sequences (e.g., Sobol sequences) [34]. In fact, assigning unseen data to the clusters serves our purpose of visualizing the decision boundaries between the clusters determined by the clustering algorithm.

Generating Feature Values for SMART and IDEA: For SMART, we evaluate a fixed data set and jointly shuffle values of the feature set S. For IDEA, we can either use observed values or strive for a more space-filling design. More values result in better FAs but higher computational costs.

Reassigning versus Reclustering: FACT aims to explain a given clustering of the data. The found clustering outcome is treated as "a snapshot in time", similarly to how explanations in SL are conditional on a trained model. FACT methods are therefore akin to model-agnostic interpretation methods in SL. It follows that we need a reassignment of instances to pre-found clusters instead of a reclustering (running the clustering algorithm from the ground up). Reclustering artificial data would result in a "concept drift" and different clusters, thus being counterproductive to our goals.

In Fig. 2 (left), we create an artificial data set using the Cartesian product of the original bivariate data that forms 3 clusters and reassign the artificially created observations to the found clusters of a cluster model fitted on the original bivariate data (grid lines). The right plot visualizes a reclustering of the same artificial data set, resulting in clearly visible changes in the shape and position of the clusters.

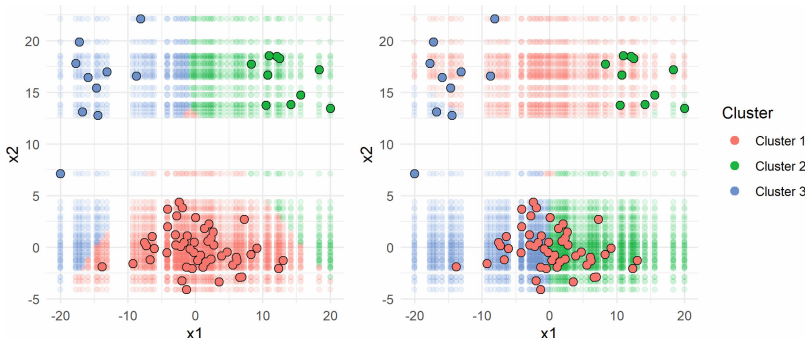


Fig. 2. Observations (solid points) and Cartesian product (transparent grid) reassigned (left plot) and reclustered (right plot).

How the FACT Framework is Algorithm-Agnostic: How to reassign instances differs across clustering algorithms. For instance, in k-means we assign an instance to the cluster with the lowest Euclidean distance; in probabilistic clustering such as Gaussian mixture models we select the cluster associated with the largest probability; in hierarchical clustering, we select the cluster with the lowest linkage value, etc. [8]. In other words, although the implementation of the reassignment stage differs across algorithms (the computation of soft or hard labels), FACT methods stay exactly the same. For FACT to be truly algorithm-agnostic, we develop variants to accommodate both soft and hard labeling algorithms.

Limitations: FACT is not suited for evaluating the quality of the clustering, i.e., whether clusters have a high within-cluster homogeneity and high between-cluster heterogeneity. Furthermore, we need an appropriate assignment function that assigns instances to existing clusters and which may frequently not be available. Particularly IDEA is limited by computational constraints for large data sets. Hence, we introduce a sampling stage for FACT, where only a subset of clustered observations can be selected to estimate FAs.

5 Simulations

5.1 Flexibility of SMART - Micro F1 versus Macro F1

In this simulation, we illustrate that the micro F1 score and therefore also the G2PC proposed in [8] is not useful for imbalanced cluster sizes. We also demonstrate the advantages of our more flexible SMART approach, which allows us to use the macro F1 score instead, a scoring metric better suited for imbalanced cluster sizes. We simulate a data set with two features consisting of 4 differently sized classes (see Fig. 3), where each class follows a different bivariate normal distribution. 60 instances are sampled from class 3 while 20 instances are sampled from each of the remaining classes. To capture the latent class variable, c-means is initialized at the 4 centers. The right plot in Fig. 3 displays the perfect cluster

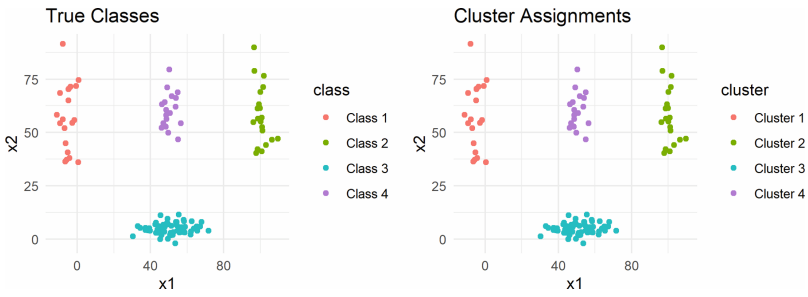


Fig. 3. Visualization of the data and the perfect clustering of c-means.

assignments found by c-means. We can see that x_1 is the defining feature of the clustering for 3 out of 4 clusters, i.e., for the clusters enumerated by 1, 2, and 4. Our goal is to analyze the c-means clustering model to discover which of the two features were more important for the clustering outcome.

We now compare the macro F1 score and micro F1 score (see Appendix B) for x_1 and x_2. Both features have micro F1 median scores of 0.58, suggesting equal importance for x_1 and x_2. Recall that the micro F1 score corresponds to 1 - G2PC (see Theorem 1). This implies that G2PC is unable to identify a meaningful feature importance ranking for x_1 and x_2 in this case. Macro F1 on the other hand is different for both features ($x_1 = 0.43, x_2 = 0.64$), indicating that x_1 is more important. Note that the F1 score is a similarity index. A low F1 score indicates a high feature importance, i.e., a high dissimilarity between the clustering outcome based on the original data and the clustering outcome after the feature of interest has been shuffled. These results stem from the fact that micro F1 accounts for each instance with equal importance (by globally counting true and false positives, see Appendix B). Cluster 3 is over-represented with three times as many instances as the remaining clusters. The macro F1 score accurately captures this by treating each cluster as equally important, regardless of its size.

5.2 Global versus Cluster-Specific SMART

Next, we demonstrate that even when using the macro F1 score for imbalanced clusters, the results may obfuscate the importance of features to specific clusters, which is where cluster-specific SMART becomes the method of choice. We simulate three visibly distinctive classes (left plot in Fig. 4) where each class follows a bivariate normal distribution with different mean and covariance matrices. 50 instances are sampled from class 2, and 20 instances are sampled from class 1 and class 3 each. We initialize c-means at the 3 mean values. As shown in Fig. 4, the cluster assignments capture all three classes almost perfectly, except for one instance of class 2 being assigned to cluster 1 and one to cluster 3.

We compare the global macro F1 (which weights the importance of clusters equally) to the cluster-specific F1 score. With a global macro F1 median of 0.62 for x_1 and 0.66 for x_2, there is no difference between the importance of both

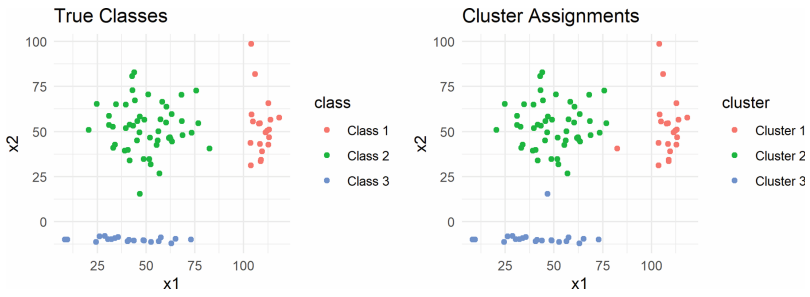


Fig. 4. Three classes with different distributions clustered by c-means. True classes (left) and clusters (right) almost perfectly match.

features for the overall clustering. In contrast, cluster-specific SMART offers a more detailed view of the contributions of each feature to the clustering outcome. Both features, x_1 and x_2, have an equal regional feature importance of 0.73 in forming cluster 2. For cluster 3, feature x_2 is considerably more important with a macro F1 score of 0.26, compared to 0.86 for feature x_1. Vice versa, feature x_1 is the defining feature of cluster 1 with a score of 0.24. In comparison, the importance of x_2 for cluster 1 is 1.0, implying that the permutation of feature x_2 had no effect on the assignment criteria for cluster 1.

5.3 How to Interpret IDEA

Here, we demonstrate how IDEA can visualize isolated, univariate effects of features on the cluster assignments of multi-dimensional data; how the heterogeneity of local effects influences the explanatory power of the global IDEA; and how grouping IDEA curves by initial cluster assignments reveals similar effects. We draw 50 instances from three multivariate normally distributed classes. To make them differentiable for the clustering algorithm, the classes are generated with an antagonistic mean structure. The covariance matrix of the three classes is sampled using a Wishart distribution (see Appendix C for details). The left plot in Fig. 5 depicts the three-dimensional distribution of the classes. We intend class 3 to be dense and classes 1 and 2 to be less dense but large in hypervolume. We initialize c-means at the 3 centers and optimize via the Euclidean distance. Figure 5 visualizes the perfect clustering. Figure 6 (left) displays an hIDEA plot for x_1 (see Sect. 3.2), indicating the majority vote of cluster assignments when exchanging values of x_1 by the horizontal axis value for all observations.

The curves in Fig. 6 (right) represent the cluster-specific components of the sIDEA function (local and global). Note that this refers to the effect of observations being reassigned to the c-th cluster and not the initial cluster effect, which we demonstrate below. The bandwidths represent the local IDEA curve ranges that were averaged to receive the respective global IDEA. We can see that - on average - x_1 has a substantial effect on the clustering outcome. The lower the value of x_1 that is plugged into an observation, the more likely it is assigned to cluster 1, while for larger values of x_1 it is more likely to be assigned to cluster

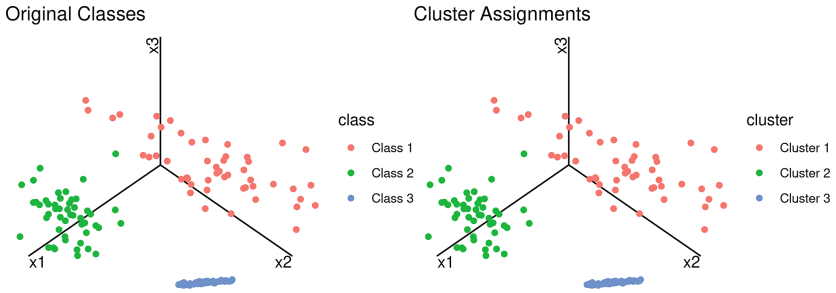


Fig. 5. Sampled classes (left plot) versus clusters (right plot).

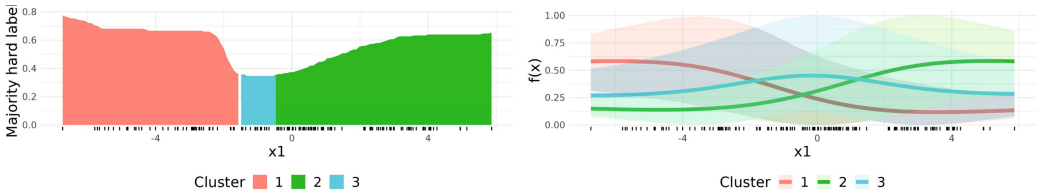


Fig. 6. Left: A plot indicating "certainty" of the global hIDEA function. On average, replacing x_1 by the axis value results in an observation being assigned to the color-indicated cluster. The vertical distance indicates how many observations are assigned to the majority cluster. **Right:** Cluster-specific global sIDEA curves. Each curve indicates the average soft label of an observation being assigned to the c-th cluster if its x_1 value is replaced by the axis value. The bandwidths visualize the distribution of local sIDEA curves that were vertically averaged to the respective global, cluster-specific sIDEA.

2. For $x_1 \approx 0$, observations are more likely to be assigned to cluster 3. The large bandwidths indicate that the clusters are spread out, and plugging in different values of x_1 into an observation has widely different effects across the data set. Particularly around $x_1 \approx 0$, where cluster 3 dominates, the average effect loses its meaning due to the underlying local IDEA curves being highly heterogeneous. In this case, one should be wary of the interpretative value of the global IDEA. We proceed to investigate the heterogeneity of the local sIDEA curves for cluster 3 (see Fig. 7 on the left). The flat shape of the cluster-specific global sIDEA indicates that x_1 has a rather low effect on observations being assigned to cluster 3. However, the cluster-specific local sIDEA curves reveal that individual effects cancel each other out when being averaged.

Initial Cluster Effect: It seems likely that observations belonging to a single cluster in the initial clustering run would behave similarly once their feature values are changed. We color each sIDEA curve by the original cluster assignment (see Fig. 7 on the right) and add the corresponding aggregate curves. Our assumption - that observations within a cluster behave similarly once we make isolated changes to their feature values - is confirmed. The formal definition of this initial cluster effect is given by Eq. (4).

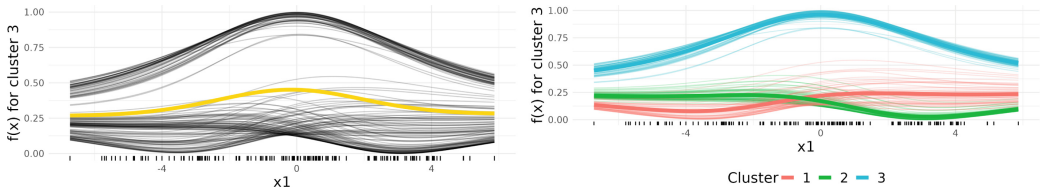


Fig. 7. Left: Cluster-specific IDEA (local and global), indicating effects on the soft labels for observations to be assigned to cluster 3. The black lines represent local effects; the yellow line the global effect. **Right:** sIDEA curves colored by initial cluster assignment. The thin curves represent local effects; the thick curves represent aggregate effects. We can see similar effects of replacing the values of x_1 on the soft labels, depending on what initial cluster an observation is part of.

5.4 IDEA Recovers Distribution Found by Clustering Algorithms

This simulation demonstrates how the global sIDEA can "recover" the distributions found by the clustering algorithm. We simulate 4 features and cluster the data into 3 clusters with FuzzyDBSCAN [20]. We illustrate soft labels for assignments to a single cluster in Fig. 8. The upper triangular plots display true bivariate marginal densities of features. The lower triangular plots display the corresponding bivariate global sIDEA estimates. Matching pairs of densities and sIDEA estimates "mirror" each other on the diagonal line. The diagonal plots visualize univariate marginal distributions (grey area) versus the corresponding estimated univariate global sIDEA curve (black line). The location and shape of sIDEA plots approximate the true marginal distributions. Note that for the correlated pairs (x_1, x_2) and (x_3, x_4), we recover the direction of the correlation.

6 Real Data Application

The Wisconsin diagnostic breast cancer (WDBC) data set [7] consists of 569 instances of cell nuclei obtained from breast mass. Each instance consists of 10 characteristics derived from a digitized image of a fine-needle aspirate. For each characteristic, the mean, standard error and "worst" or largest value (mean of the three largest values) is recorded, resulting in 30 features of the data set. Each nucleus is classified as malignant (cancer, class 1) or benign (class 2). We cluster the data using Euclidean optimized c-means. Figure 9 visualizes the projection of the data onto the first two PCs. The clusters cannot be separated with two PCs, and the visualization is of little help in understanding the influence of the original features on the clustering outcome.

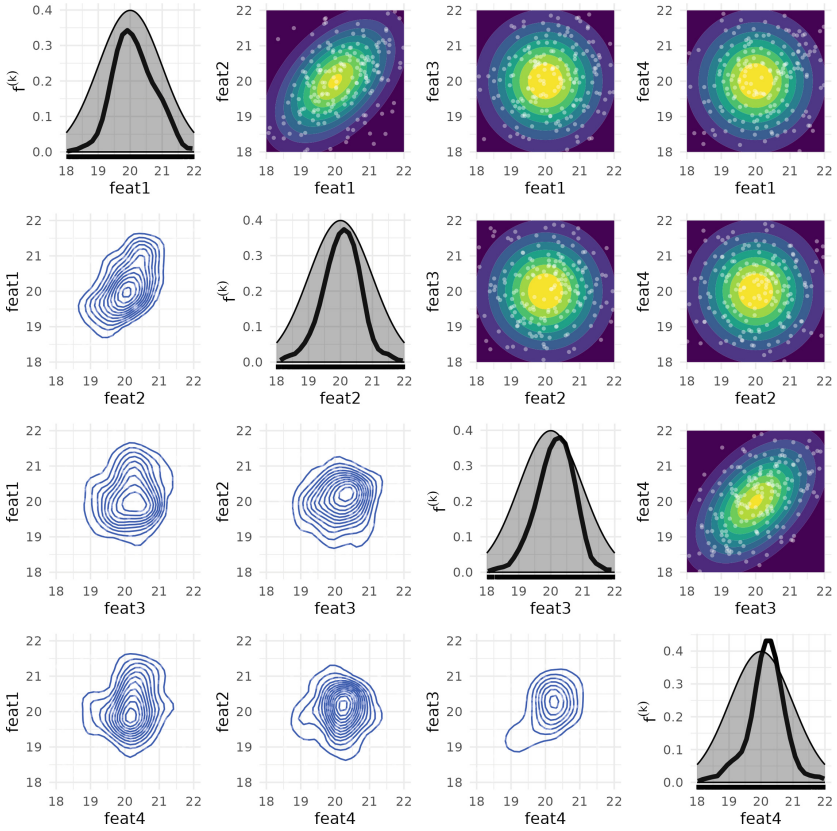


Fig. 8. Comparison of true bivariate marginal densities of features (upper triangular plots) with corresponding global bivariate sIDEA (lower triangular plots) and true univariate marginal densities of features (diagonal plots, grey area) with corresponding global univariate sIDEA (diagonal plots, black line). (Color figure online)

6.1 Aggregate FA for Each Cluster (SMART)

We first showcase how SMART can serve as an approximation of the actual reclustering. Measured on the latent target variable, the initial clustering run has an F1 score of 0.88. We then recluster the data, once with the 4 most important and once with the 4 least important features. Dropping the 26 least important features only reduces the F1 score by 0.03 to 0.85 (measured using the latent target). In contrast, using the 4 least important features reduces the F1 score by 0.55 to 0.33 and thus alters the clustering in a major way. This demonstrates that assigning new instances to existing clusters can serve as an efficient method for feature selection. To showcase the grouped feature importance, we jointly shuffle features and compare their importance in Fig. 10. Note that we use the natural logarithm of SMART here for better visual separability and to receive a natural ordering of the feature importance (due to F1 being a similarity index), where a larger bar indicates a higher importance and vice versa.

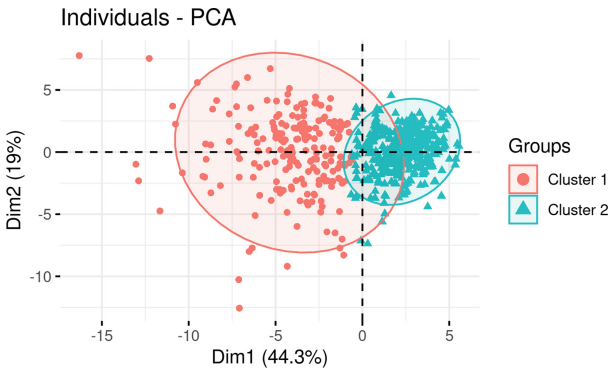


Fig. 9. First and second PCs of WDBC data with clusters of real target values.

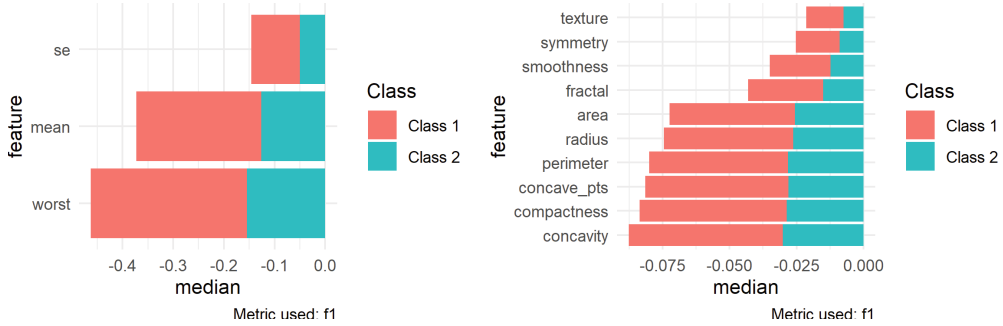


Fig. 10. Grouped SMART (using the natural logarithm) per cluster for groups of categories (left plot) and groups of characteristics (right plot) in the WDBC data set.

6.2 Visualizing Marginal Feature Effects (IDEA)

We now visualize isolated univariate and bivariate effects of features on assignments. Figure 11 plots the global IDEA curve for three features `concavity_worst`, `compactness_worst`, and `concave_points_worst`. The transparent areas indicate the regions where the local curve mass is located. A rug on the horizontal axis shows the distribution of the corresponding feature. For all three features, larger values result in observations being assigned to cluster 1, while lower values result in observations being assigned to cluster 2. The distribution of cluster-specific local IDEA curves is wide, reflecting voluminous clusters. All features have a strong univariate effect on the cluster assignments, which indicates a large importance of each feature to the constitution of each cluster.

Figure 11 (right) plots the two-dimensional sIDEA for `compactness_worst` and `compactness_mean`. The color indicates what cluster the observations are assigned to on average when `compactness_worst` and `compactness_mean` are replaced by the axis values. The transparency indicates the magnitude of the soft label, i.e., the "certainty" in our estimate. On average, the observations are assigned to cluster 2 when adjusting both features to lower values and to cluster 1 when adjusting both features to higher values.

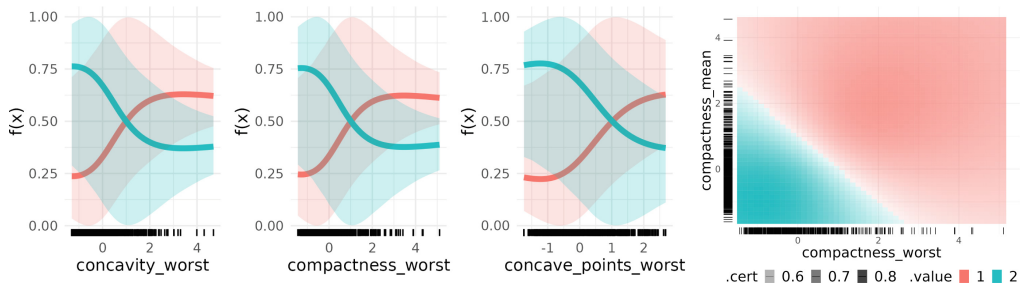


Fig. 11. Left: Univariate global sIDEA plots for the features `concavity_worst`, `compactness_worst`, and `concave_points_worst`. **Right:** Two-dimensional sIDEA for the features `compactness_worst` and `compactness_mean`. On average, an observation is assigned to cluster 1 for large values of both features, while it is assigned to cluster 2 for low values of both features.

7 Conclusion

This research paper proposes FACT, a framework to produce FAs which is compatible with any clustering algorithm able to reassign instances through soft or hard labels, preserves the integrity of the data, and does not introduce additional models. FACT techniques provide information regarding the importance of features for assigning instances to clusters (overall and to specific clusters); or how isolated changes in feature values affect the assignment of single instances or the entire data set to each cluster. We introduce two novel FACT methods: SMART and IDEA. SMART is a general framework that outputs a single global value for each feature indicating its importance to cluster assignments or one value for each cluster (and feature). IDEA adds to these capabilities by visualizing the structure of the feature influence on cluster assignments across the feature space for single observations and the entire data set.

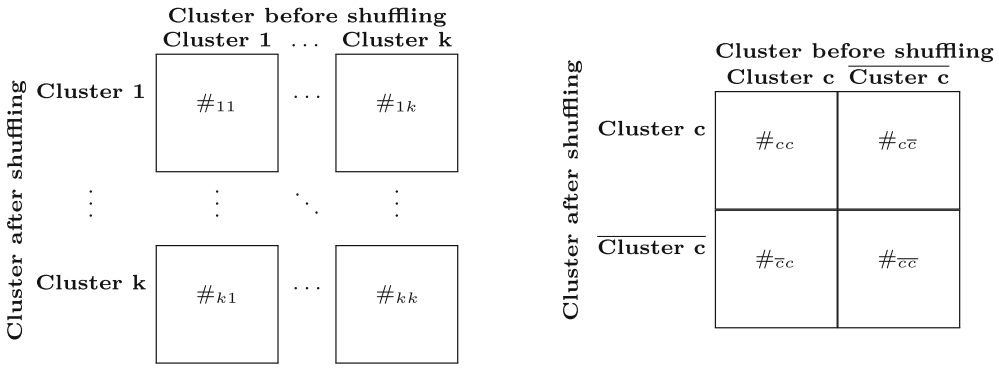
Although explaining algorithmic decisions is an active research topic in SL, it is largely ignored for clustering algorithms. The FACT framework provides a new impetus for algorithm-agnostic interpretations in clustering. With SMART and IDEA, we hope to establish a foundation for the future development of FACT methods and spark more research in this direction.

A Confusion Matrix for SMART

Transferring the concept of confusion matrices from classification tasks, a "true" classification would correspond to an observation staying within the same cluster after the intervention, and a "false" classification would result in a reassignment to a different cluster.

For the multi-cluster matrix on the left, let TP denote the sum of all true positives from all binary comparisons of cluster c versus the remaining clusters, FP the sum of all false positives, and FN the sum of all false negatives. It follows that $\sum_{l=1}^{k} \#_{ll} = \text{TP}$ and $n - \sum_{l=1}^{k} \#_{ll} = \text{FP} = \text{FN}$.

Table 1. Multi-cluster and binary confusion matrices for SMART.



For the binary matrix on the right, let TP_c denote all true positives of cluster c versus the remaining clusters, FP_c all false positives, FN_c all false negatives, and TN_c all true negatives. It follows that $\#_{cc} = \mathrm{TP}_c$, $\#_{c\bar{c}} = \mathrm{FP}_c$, $\#_{\bar{c}c} = \mathrm{FN}_c$, and $\#_{\bar{c}\bar{c}} = \mathrm{TN}_c$.

B Scores

F_β score: Balances false positives and false negatives. The F_β score of cluster c versus the remaining ones corresponds to:

$$F_{\beta,c} = \frac{\left(\beta^2+1\right)\cdot P_c \cdot R_c}{\beta^2 \cdot P_c + R_c}, \quad \text{where} \quad P_c = \frac{\#_{cc}}{\#_{cc}+\#_{\bar{c}c}} \quad \text{and} \quad R_c = \frac{\#_{cc}}{\#_{cc}+\#_{c\bar{c}}}$$

The F_1 (which we refer to as F1) score simplifies to:

$$F_{1,c} = 2\frac{P_c \cdot R_c}{P_c + R_c}$$

Given a multi-cluster confusion matrix M, let ϕ_c be an arbitrary binary scoring function dependent on TP, FP, FN, and TN. $\mathcal{S}_{\text{macro}}$ denotes the multi-cluster macro score that treats each cluster with equal importance. $\mathcal{S}_{\text{micro}}$ denotes the multi-cluster micro score that treats each instance with equal importance:

$$\mathcal{S}_{\text{macro}}(M) = \frac{1}{k}\sum_{c=1}^{k} \phi\left(\mathrm{TP}_c, \mathrm{FP}_c, \mathrm{FN}_c, \mathrm{TN}_c\right)$$

$$\mathcal{S}_{\text{micro}}(M) = \phi\left(\sum_{c=1}^{k}\mathrm{TP}_c, \sum_{c=1}^{k}\mathrm{FP}_c, \sum_{c=1}^{k}\mathrm{FN}_c, \sum_{c=1}^{k}\mathrm{TN}_c\right)$$

C Wishart Distribution

We sample the covariance matrix M from the Wishart distribution with $M \sim \mathrm{Wishart}_3(3, \Sigma)$. Σ is constructed using $\Sigma_{\text{Class 1}} = 0.6I_3$, $\Sigma_{\text{Class 2}} = 0.3I_3$, and

$\Sigma_{\text{Class 3}} = 0.15 I_3$, where I_3 refers to the 3×3 identity matrix. As a result, the variance of class 1 is the largest, the variance of class 3 is the lowest, and the variance of class 2 lies between the variances of classes 1 and 3.

D Proofs

Proof (Theorem 1).

Recall the definition of G2PC with respect to a multi-cluster confusion matrix M (see Table 1 in Appendix A):

$$\text{G2PC}(M) = \frac{1}{n}\left(\sum_{i=1}^{k}\sum_{j=1}^{k} \#_{ij} - \sum_{l=1}^{k} \#_{ll}\right) = \frac{1}{n}\left(n - \sum_{l=1}^{k} \#_{ll}\right) = 1 - \frac{1}{n}\sum_{l=1}^{k} \#_{ll}$$

Let TP denote the number of true positive labels, FP the number of false positives, and FN the number of false negatives. The sum of diagonal elements corresponds to TP:

$$\sum_{l=1}^{k} \#_{ll} = \text{TP}$$

It follows that:

$$\text{G2PC}(M) = 1 - \frac{\text{TP}}{n}$$

TP divided by the absolute number of instances equals the percentage of "correctly classified instances" (the number of instances staying within the same cluster after the intervention in our case) which corresponds to accuracy (ACC):

$$\frac{1}{n}\sum_{l=1}^{k} \#_{ll} = \frac{\text{TP}}{n} = \text{ACC}(M)$$

It follows that:

$$\text{G2PC}(M) = 1 - \text{ACC}(M) \Leftrightarrow 1 - \text{G2PC}(M) = \text{ACC}(M) \tag{5}$$

The following relation holds by definition for the micro F1 score [38]:

$$\text{F1}_{\text{micro}}(M) = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})}$$

For multi-class classification it holds that FP = FN, as every false positive for one class is a false negative for another class. With $n = \text{TP} + \text{FP}$, it follows that:

$$\text{F1}_{\text{micro}}(M) = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{\text{TP}}{n} = \text{ACC}(M) \tag{6}$$

From Eqs. (5) and (6), we have:

$$1 - \text{G2PC}(M) = \text{F1}_{\text{micro}}(M)$$

□

References

1. Achtert, E., Böhm, C., Kriegel, H.P., Kröger, P., Zimek, A.: Deriving quantitative models for correlation clusters. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2006, pp. 4–13. Association for Computing Machinery, New York, NY, USA (2006)
2. Apley, D.W., Zhu, J.: Visualizing the effects of predictor variables in black box supervised learning models. J. R. Stat. Soc. Ser. B **82**(4), 1059–1086 (2020)
3. Bertsimas, D., Orfanoudaki, A., Wiberg, H.: Interpretable clustering via optimal trees. ArXiv e-prints (2018). arXiv:1812.00539
4. Bertsimas, D., Orfanoudaki, A., Wiberg, H.: Interpretable clustering: an optimization approach. Mach. Learn. **110**(1), 89–138 (2021)
5. Blockeel, H., Raedt, L.D., Ramon, J.: Top-down induction of clustering trees. In: Proceedings of the Fifteenth International Conference on Machine Learning, ICML 1998, pp. 55–63. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1998)
6. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
7. Dua, D., Graff, C.: UCI machine learning repository (2019). http://archive.ics.uci.edu/ml
8. Ellis, C.A., Sendi, M.S.E., Geenjaar, E.P.T., Plis, S.M., Miller, R.L., Calhoun, V.D.: Algorithm-agnostic explainability for unsupervised clustering. ArXiv e-prints (2021). arXiv:2105.08053
9. 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)
10. Fraiman, R., Ghattas, B., Svarc, M.: Interpretable clustering using unsupervised binary trees. Adv. Data Anal. Classif. **7**(2), 125–145 (2013)
11. Friedman, J.H.: Greedy function approximation: a gradient boosting machine. Ann. Stat. **29**(5), 1189–1232 (2001)
12. Frost, N., Moshkovitz, M., Rashtchian, C.: ExKMC: Expanding explainable k-means clustering. ArXiv e-prints (2020). arXiv:2006.02399
13. Funk, H., Scholbeck, C.A., Casalicchio, G.: FACT: Feature Attributions for Clus-Tering (2023). https://CRAN.R-project.org/package=FACT. R package version 0.1.0
14. Gabidolla, M., Carreira-Perpiñán, M.A.: Optimal interpretable clustering using oblique decision trees. In: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2022. pp. 400–410. Association for Computing Machinery, New York, NY, USA (2022)
15. Ghattas, B., Michel, P., Boyer, L.: Clustering nominal data using unsupervised binary decision trees: comparisons with the state of the art methods. Pattern Recognit. **67**, 177–185 (2017)
16. 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)
17. Hinneburg, A.: Visualizing clustering results. In: Liu, L., Özsu, M.T. (eds.) Encyclopedia of Database Systems, pp. 3417–3425. Springer, Boston (2009). https://doi.org/10.1007/978-0-387-39940-9_617
18. Hooker, G.: Generalized functional anova diagnostics for high-dimensional functions of dependent variables. J. Comput. Graph. Stat. **16**(3), 709–732 (2007)

19. 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. Stat. Comput. **31**(6), 82 (2021)
20. Ienco, D., Bordogna, G.: Fuzzy extensions of the DBScan clustering algorithm. Soft. Comput. **22**(5), 1719–1730 (2018)
21. Jaccard, P.: The distribution of the flora in the alpine zone. New Phytol. **11**(2), 37–50 (1912)
22. Kinkeldey, C., Korjakow, T., Benjamin, J.J.: Towards supporting interpretability of clustering results with uncertainty visualization. In: EuroVis Workshop on Trustworthy Visualization (TrustVis) (2019)
23. Lawless, C., Kalagnanam, J., Nguyen, L.M., Phan, D., Reddy, C.: Interpretable clustering via multi-polytope machines. ArXiv e-prints (2021). arXiv:2112.05653
24. Liu, B., Xia, Y., Yu, P.S.: Clustering through decision tree construction. In: Proceedings of the Ninth International Conference on Information and Knowledge Management, CIKM, pp. 20–29. Association for Computing Machinery, New York, NY, USA (2000)
25. Loyola-González, O., et al.: An explainable artificial intelligence model for clustering numerical databases. IEEE Access **8**, 52370–52384 (2020)
26. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS 2017, pp. 4768–4777. Curran Associates Inc., Red Hook, NY, USA (2017)
27. Molnar, C.: Interpretable Machine Learning (2019). https://christophm.github.io/interpretable-ml-book/
28. Molnar, C., Casalicchio, G., Bischl, B.: Interpretable machine learning - a brief history, state-of-the-art and challenges. In: Koprinska, I., et al. (eds.) ECML PKDD 2020 Workshops, pp. 417–431. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-65965-3_28
29. Molnar, C., et al.: General pitfalls of model-agnostic interpretation methods for machine learning models. In: Holzinger, A., Goebel, R., Fong, R., Moon, T., Müller, K.R., Samek, W. (eds.) xxAI 2020. LNCS, vol. 13200, pp. 39–68. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-04083-2_4
30. Moshkovitz, M., Dasgupta, S., Rashtchian, C., Frost, N.: Explainable k-means and k-medians clustering. In: III, H.D., Singh, A. (eds.) Proceedings of the 37th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 119, pp. 7055–7065. PMLR (2020)
31. Plant, C., Böhm, C.: INCONCO: interpretable clustering of numerical and categorical objects. In: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2011, pp. 1127–1135. Association for Computing Machinery, New York, NY, USA (2011)
32. Rand, W.M.: Objective criteria for the evaluation of clustering methods. J. Am. Stat. Assoc. **66**(336), 846–850 (1971)
33. 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, KDD 2016, pp. 1135–1144. Association for Computing Machinery, New York, NY, USA (2016)
34. Saltelli, A., et al.: Global Sensitivity Analysis: The Primer. John Wiley & Sons Ltd, Chichester (2008)
35. 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
36. Sobol, I.: Global sensitivity indices for nonlinear mathematical models and their monte carlo estimates. Math. Comput. Simul. **55**(1), 271–280 (2001)
37. Strumbelj, E., Kononenko, I.: An efficient explanation of individual classifications using game theory. J. Mach. Learn. Res. **11**, 1–18 (2010)
38. Takahashi, K., Yamamoto, K., Kuchiba, A., Koyama, T.: Confidence interval for micro-averaged F1 and macro-averaged F1 scores. Appl. Intell. **52**(5), 4961–4972 (2022)
39. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: automated decisions and the GDPR. Harvard J. Law Technol. **31**(2) (2018)