Moritz Herrmann

Towards more reliable machine learning: conceptual insights and practical approaches for unsupervised manifold learning and supervised benchmark studies

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

Eingereicht am 02.08.2022

Moritz Herrmann

Towards more reliable machine learning: conceptual insights and practical approaches for unsupervised manifold learning and supervised benchmark studies

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

Eingereicht am 02.08.2022

Erster Berichterstatter: PD Dr. Fabian Scheipl
Zweiter Berichterstatter: Prof. Dr. Sonja Greven
Dritter Berichterstatter: Prof. Dr. Christian L. Müller

Tag der Disputation: 28.10.2022

Acknowledgments

First and foremost, I would like to thank my advisor PD Dr. Fabian Scheipl for taking me over as a doctoral candidate, his extraordinary and outstanding support, advice, and encouragement. I am deeply grateful for the freedom he gave me to conduct my research and his patience and support in that.

Moreover, I thank ...

... Prof. Dr. Sonja Greven and Prof. Dr. Christian L. Müller for taking over the roles of the second and third reviewer of my Ph.D. thesis.

... Prof. Dr. Christian Heumann and Prof. Dr. Thomas Augustin for completing the doctoral committee.

... Prof. Dr. Anne-Laure Boulesteix for her support in finishing the survival benchmark paper and her continuing advice.

... Prof. Dr. Bernd Bischl for including me in his "extended" working group.

... Prof. Dr. Peer Kröger for the easy collaboration.

... Brigitte Maxa and Elke Höfner for their help in organizational matters.

I would also like to thank ...

... all my coauthors for the good and fruitful cooperation. In particular, I would like to express my gratitude to Daniyal Kazempour and Florian Pfisterer for the many advices and support in all the things to consider besides the pure scientific work.

... all my colleagues and friends at the Department of Statistics. I would especially like to thank Daniel, Julia, and Jann for all the fun in the past years.

... Martin for the great cooperation in "Programming with R" and the many interesting conversations.

Last but not least, I would like to thank my close friends and family. This thesis would not have been possible without you making life beautiful. In particular, I would like to thank my mother Katja, my grandmother Brigitte, and my brother Johannes. Most of all, I am deeply grateful to my girlfriend Elena. Your support, patience and love means everything to me.

Summary

The results of methodological research in computational statistics and machine learning are increasingly called into question concerning their general replicability, reliability, and trustworthiness. Thus, the fundamental guiding principle of this thesis is to work towards an improvement of these important aspects of machine learning. While assessing the reliability of supervised learning methods is already relatively well researched and established in the form of benchmark studies, the question of reliability and trustworthiness in unsupervised learning is much more involved, mainly because evaluating methods in unsupervised learning is very difficult since there is usually no "ground truth" to compare the results against.
The main focus of this work is to elaborate on a better understanding of the underlying conceptual principles and towards improved practical reliability of manifold learning. Manifold learning or nonlinear dimension reduction is concerned with learning low-dimensional representations that faithfully reflect the intrinsic structure of ostensible high-dimensional and complex data. Finding such embeddings is of great importance for data exploration, visualization, and interpretable analysis. However, how to assess and evaluate whether such embeddings faithfully and reliably reflect the intrinsic structure is a fundamental open problem and manifold learning is prone to overoptimistic and unreliable findings. In particular, many manifold learning methods depend on several hyperparameters that substantially affect the embedding results. First of all, it is thus investigated whether and how reliably existing embedding evaluation measures can be used for tuning manifold learning methods. Secondly, outlier detection and cluster analysis are investigated from a manifold learning perspective. First, a general conceptualization of outlier detection as a geometrical problem is developed within the specific context of functional data. It is demonstrated with extensive experiments that simple, well-established manifold learning methods in combination with standard outlier detection methods can improve conceptual understanding and practical feasibility of functional outlier detection. The proposed ideas are then generalized to other high-dimensional and non-tabular data types such as graphs and images. In a similar vein, cluster analysis is considered from a topological perspective. Concepts from topological data analysis and manifold learning are brought together to improve the understanding of principles underlying the problem of cluster analysis. The recently proposed manifold learning method UMAP is used to infer the topological structure of a data set and the well-established method DBSCAN for cluster detection. Extensive experiments with simulated and real data show that exploiting the topological structure of a data set before clustering can considerably improve cluster analysis. In addition to this main contribution, the thesis also contributes to reliability in supervised learning. First of all, it includes an example of a benchmark study focusing on survival prediction methods in multi-omics cancer data. Moreover, it describes a follow-up study that assesses the effects of the multiple design and analysis options on the results of benchmark studies.

Zusammenfassung

Die Ergebnisse methodischer Forschung im Bereich der computergestützten Statistik und des maschinellen Lernens werden hinsichtlich ihrer Reproduzierbarkeit, Zuverlässigkeit und Vertrauenswürdigkeit zunehmend in Frage gestellt. Daher ist der grundlegende Leitgedanke dieser Arbeit, auf eine Verbesserung dieser essentiellen Aspekte des maschinellen Lernens hinzuarbeiten. Während Ansätze zur Bewertung der Zuverlässigkeit und Reproduzierbarkeit von Ergebnissen im überwachten Lernen bereits relativ gut erforscht und in Form von Benchmark-Studien etabliert sind, ist die Frage nach der Zuverlässigkeit und Vertrauenswürdigkeit von unüberwachtem Lernen sehr viel komplizierter. Das liegt vor allem daran, dass es beim unüberwachten Lernen in der Regel keine "Grundwahrheit" gibt, mit der die Ergebnisse der unüberwachten Methoden verglichen werden können.
Das Hauptaugenmerk dieser Arbeit liegt auf einem besseren Verständnis der zugrunde liegenden konzeptionellen Prinzipien und auf einer verbesserten praktischen Verlässlichkeit des Manifold Learning. Beim Manifold Learning (auch nichtlineare Dimensionsreduktion genannt) geht es darum, niedrigdimensionale Repräsentationen zu finden, die die intrinsische Struktur vermeintlich hochdimensionaler und komplexer Daten getreu wiedergeben. Solche Einbettungen sind für die Exploration, Visualisierung und interpretierbare Analyse von Daten von großer Bedeutung. Wie jedoch beurteilt und bewertet werden kann, ob solche Einbettungen die innere Struktur getreu und zuverlässig widerspiegeln, ist ein grundlegendes, offenes Problem, und Manifold Learning ist anfällig für überoptimistische und unzuverlässige Ergebnisse. Insbesondere hängen viele Methoden von verschiedenen Hyperparametern ab, die die Ergebnisse erheblich beeinflussen. Zunächst wird daher untersucht, ob und wie zuverlässig bestehende Evaluationsmaße für die Spezifizierung von Hyperparametern von Manifold-Learning-Methoden verwendet werden können. Zudem werden Ausreißererkennung und Clusteranalyse aus einer Manifold-Learning-Perspektive untersucht. Zunächst wird eine allgemeine Konzeptualisierung der Ausreißererkennung als geometrisches Problem im spezifischen Kontext von funktionalen Daten entwickelt. Mit umfangreichen Experimenten wird gezeigt, dass einfache, gut etablierte Manifold-Learning-Methoden in Kombination mit Standardmethoden zur Ausreißererkennung das konzeptionelle Verständnis und die praktische Durchführbarkeit der funktionalen Ausreißererkennung verbessern können. Die vorgeschlagenen Ideen werden dann auf andere hochdimensionale und nicht-tabellarische Datentypen wie Graphen und Bilder verallgemeinert. In ähnlicher Weise wird die Clusteranalyse aus einer topologischen Perspektive betrachtet. Konzepte aus der topologischen Datenanalyse und dem Manifold Learning werden zusammengeführt, um das Verständnis der Prinzipien, die dem Problem der Clusteranalyse zugrunde liegen, zu verbessern. Die kürzlich vorgeschlagene Manifold-Learning-Methode UMAP wird verwendet, um die topologische Struktur eines Datensatzes zu ermitteln, und die etablierte Methode DBSCAN für die Clustererkennung. Umfangreiche Experimente mit simulierten und realen Daten zeigen, dass das Herausarbeiten der topologischen Struktur eines Datensatzes vor dem Clustering die Clusteranalyse erheblich verbessern kann.
Neben diesem Hauptbeitrag leistet die Arbeit auch einen Beitrag zur Zuverlässigkeit beim überwachten Lernen. Zunächst enthält sie ein Beispiel einer Benchmark-Studie, die sich auf Überlebenszeitvorhersagemethoden in Multi-Omics-Krebsdaten konzentriert. Darüber hinaus wird eine Folgestudie beschrieben, in der die Auswirkungen der verschiedenen Design- und Analyseoptionen auf die Ergebnisse von Benchmark-Studien untersucht werden.

Contents

Part I.

Introduction and Background

1. Introduction

> So we really ought to look into theories that don't work, and science that isn't science.
> — Richard Feynman

1.1. Motivation and Scope

In Caltech's 1974 commencement address, Richard Feynman coined the term *Cargo Cult Science*. He described certain practices and habits he had observed among researchers to adhere to a certain form, but which he believed were contrary to the basic principles of the scientific method. Among other things, he stated that "if you're doing an experiment, you should report everything that you think might make it invalid—not only what you think is right about it" and that if "you've made up your mind to test a theory, or you want to explain some idea, you should always decide to publish it whichever way it comes out" (Feynman, 1974).

In 2005, John Ioannidis showed "Why Most Published Research Findings Are False" (Ioannidis, 2005, p. 0698). For example, he outlined that "the hotter a scientific field (with more scientific teams involved)" and "the greater the flexibility in designs, definitions, outcomes, and analytical modes in a scientific field, the less likely the research findings are to be true".

In 2016, Monya Baker reported that of 1576 researchers who responded to a *Nature* online survey, 70 % could not reproduce the results of others and over 50 % could not reproduce their own results. *Selective reporting*, *pressure to publish*, and *low statistical power or poor analysis* are mentioned most often as the driving forces contributing to irreproducibility (Baker, 2016).

Now there are increasing warnings that methodological research in computational sciences and artificial intelligence also faces a replication crisis (Boulesteix et al., 2020; Hutson, 2018) and a large number of terms have been established to describe practices that can also be considered to fall under the umbrella term *Cargo Cult Science*: *p-hacking*, *data dredging*, *hypothesizing after the results are known (HARKing)*, or *state-of-the-art (SotA) hacking* are examples of procedures consciously or unconsciously conducted by researchers to meet the form of reporting positive results, but which eventually lead to overoptimistic findings and making research unreliable to a large extent (Gencoglu et al., 2019; Munafò et al., 2017).

Even though there is still a lack of *neutral comparison* and *replication studies* and a *publication bias* favoring positive research findings (Boulesteix et al., 2020; Munafò et al., 2017), a growing body of empirical evidence from systematic benchmark studies supports the warnings of a replication crisis in machine learning research and suggests that there are also *Cargo Cult Science* practices taking place to some extent. Reported methodological improvements, which often amount to proposed novel methods outperforming existing ones in terms of some (prediction) performance metric, cannot be confirmed if assessed systematically in independent

comparison studies, for example, for research on generative adversarial networks (Lucic et al., 2018), deep reinforcement learning (Henderson et al., 2018), machine translation (Marie et al., 2021), or in bioinformatics (Buchka et al., 2021). Without assuming bad faith on the part of individuals, there is little other conclusion than that institutional pressures and dysfunctional routines and standards guide researchers toward conducting these practices at least unconsciously.

In conclusion, this raises serious concerns about the reliability and trustworthiness of methodological results and calls into question the scientific progress in the field of machine learning to some extent (Van Mechelen et al., 2018; Zimmermann, 2020). This is why this thesis aims to work towards conceptual insights and practical approaches to improving reliability in machine learning research. As will be outlined in the remainder of Part I, there are a variety of general and domain-specific facets to the problem so one has to focus on certain aspects.

Here the main focus lies on manifold learning as an approach for unsupervised learning and dimensionality reduction. In addition to this main contribution, the thesis also provides some insights into supervised benchmarking studies within two articles that are not concerned with manifold learning. That said, it should be emphasized: proposing novel computational algorithms or learning methods is not in the scope of this work and only established methods introduced elsewhere are used. In contrast, the thesis contributes to more reliability as follows: (1) sharpening conceptual and methodological underpinnings and (2) providing extensive practical evaluations and comparisons of existing methods. If the general assumption is that learning methods create theories to explain data (Wolpert, 2020), the basic goal of this work can be summarized in the words of Richard Feynman as to look into theories about data that do not work.

1.2. Outline and Contributions

This cumulative thesis is divided into eleven chapters in four parts. The remainder of Part I builds the basis for a comprehensive understanding and provides the background on the considered topics. Chapter 2 first specifies what is understood under *reliability in machine learning* in general. Chapter 3 then discusses *reliability* in the context of *supervised learning* with a focus on *benchmark studies*. Chapter 4 finally gives an overview of the relevant aspects of *manifold learning* and the issue of *reliability* in *unsupervised learning*.

Part II and III represent the main body of the thesis and include the six contributing papers in Chapters 5 - 10. Part II is devoted to supervised benchmark studies and includes Chapter 5, which presents such a benchmark study in the context of high-dimensional multi-omics data, and Chapter 6, which describes a follow-up study substantially expanding the analysis of the results presented in Chapter 5.
Part III then includes four papers on unsupervised manifold learning. Chapter 7 first investigates tuning approaches of manifold learning methods in the context of functional data. Chapters 8 and 9 then focus on outlier detection and consider the problem from a manifold learning perspective. Chapter 8 again has a specific focus on functional data, while Chapter 9 generalizes the proposed ideas for outlier detection to other data types such as graphs and images. Chapter 10 finally draws on connections from topological data analysis, manifold learning, and density-based clustering to provide conceptual insights and practical approaches to enhance cluster detection.

Part IV finally concludes the thesis and points out future research directions.

2. Definition of Terms

> Every genuine test of a theory is an attempt to falsify it, or to refute it.
> — Karl Popper

In this thesis, the term *reliability* is used to refer to the trustworthiness of a study in a broad sense, i.e., that others can, in general, rely on the results, findings, and conclusions reached. However, this requires some discussion. First of all, it means *reliability* is closely related to aspects often described under the rather well-established terms *reproducibility* and *replicability* and there should be justification for deviating from this terminology. Secondly, *reliability* has a specific meaning in other research areas. For example, in quantitative research fields such as medicine, psychology, or educational studies, *reliability* describes the "extent to which the results are consistent if the study would be replicated" (Frambach et al., 2013) and there have been attempts to transfer this quality criterion (together with *validity* and *objectivity*) to machine learning (Myrtveit et al., 2005; Segebarth et al., 2020). Moreover, note that there are several other terms in machine learning research that describe similar or related aspects, for example, *credibility* (D'Amour et al., 2020; Marie et al., 2021), *overoptimism* (Boulesteix, 2010; Jelizarow et al., 2010), or *comparability* (Aßenmacher & Heumann, 2020; Klemenjak et al., 2020). In general, there is no established and generally agreed-upon terminology in machine learning to describe the topic. Certainly, the most frequently used terms in this regard are *reproducibility* and *replicability* and Barba (2018) provides a systematic overview of these aspects, pointing out that these terms are not used consistently (see also Plesser, 2018). Moreover, they are not sufficient for our purpose. Therefore, *reliability* is understood here as a three-level concept including *reproducibility* and *replicability*, but also an aspect we call *conceptual reliability*, specified as follows.

Reproducibility of Computational Research Results *Reproducibility* is understood as "authors provide all the necessary data and the computer codes to run the analysis again, re-creating the results" (Barba, 2018, p. 3). Note that Tatman et al. (2018) further differentiate *low*, *medium*, and *high* reproducibility. Low reproducibility means that a result should be reproducible by other researchers solely by the specification provided in the respective paper. Raff (2019) calls this *independent* reproducibility because reproducibility does not depend on the availability of code and data. In contrast, medium reproducibility requires providing the code and data, and high reproducibility additionally the complete software environment (including, for example, all dependencies) employing virtual machines, containers, or hosting services. However, meeting these reproducibility requirements is not the standard. For example, Raff (2019) found that 63.5 % of the 255 considered papers not to be independently reproducible. Moreover, at NIPS less than 40 % of papers provided code in 2017 (Tatman et al., 2018) and less than 50 % in 2018 (Pineau et al., 2020). After changing the code submission policy, 74.4 % of papers provided code at camera-ready in 2019 (Pineau et al., 2020). Consequently, further fostering all levels of reproducibility is very important and there are increasing efforts in this direction (Heil et al., 2021; Pineau et al., 2020). However, technical complexity (Sculley et al., 2015) and ever-growing models requiring

increasing amounts of computation and storage resources (Aßenmacher et al., 2021), make reproducibility a non-trivial task even if a *high* level of reproducibility is met.

Replicability Despite its importance, reproducibility is not sufficient for reliability as understood here. Reproducibility as considered above is limited to technical aspects. In contrast to that, *replicability* here means that studies "arrive at the same scientific findings as another study, collecting new data (possibly with different methods) and completing new analyses" (Barba, 2018, p. 3). Note, some authors are using the terms *reproducibility* and *replicability* in quite the opposite way (Drummond, 2009). The crucial aspect is that *replicability* does not mean that the results of a study are exactly reproduced, but the same conclusions can be drawn from a different study, in particular on different data. In terms of reliability, this makes *replicability* more important than *reproducibility*. In machine learning research, *replicability* is particularly relevant when it comes to the question of method superiority, for example, in terms of prediction performance. Supervised benchmark studies as considered in Part II are approaches to systematically compare methods proposed elsewhere given a larger and more diverse selection of benchmark data sets than the data the methods were initially developed and evaluated on. They are thus tools to improve replicability in machine learning research. If the choice of data sets follows strict inclusion criteria and the number of included data sets is large enough, the results may also be *generalizable* in terms of *null hypothesis significance testing* (NHST) (see Boulesteix et al., 2017). Note, given this definition, *replicability* appears closely related to *reliability* as used in quantitative research fields, where it is defined as the "extent to which the results are consistent if the study would be replicated" (Frambach et al., 2013).

Conceptual Clarity We would argue, however, that *replicability* does not sufficiently reflect an aspect we refer to as *conceptual clarity* (cf. *generalizability*, Arnold et al., 2019; Pineau et al., 2020). In some areas of machine learning research, there appears to be a fundamental vagueness or ambiguity about the concepts of interest. To the best of our knowledge, this aspect has not yet been described and demonstrated comprehensively and clearly enough and it is best illustrated with a few examples.

In their recent overview on outlier detection, Zimek & Filzmoser (2018, p. 7) devote a complete section to the epistemological question of what constitutes an outlier, discuss several vague and contradicting definitions in the literature, and emphasize that there are "different types of data objects occasionally termed 'outliers' ". This may be the reason for Unwin (2019, p. 635) to state: "Outliers are a complicated business. It is difficult to define what they are, it is difficult to identify them, and it is difficult to assess how they affect analyses". In a similar vein, Shalev-Shwartz & Ben-David (2014, p. 307) define clustering as "the task of grouping a set of objects such that similar objects end up in the same group and dissimilar objects are separated into different groups", only to emphasize that "this description is quite imprecise and possibly ambiguous" and that, "surprisingly, it is not at all clear how to come up with a more rigorous definition". In manifold learning, Y. Wang et al. (2021, pp. 1, 4) outline that there "are two primary types of approaches to DR [dimension reduction] for visualization, commonly referred to as local and global methods", but that there is also "no single definition of what it means to preserve local or global structure", and that "the choice of which components to preserve is important". And regarding comparisons in supervised learning, Hand (2006, p. 12) states that the considered collection of real data sets "will not be representative of real data sets in any formal sense".

To rely upon results, findings, and conclusions of a study, it appears inevitable to have a

clear understanding of the underlying concepts the study builds upon, for example, what an outlier is or what cluster detection exactly means. Without clear conceptualizations, this understanding will be hard to reach. Consequently, conceptual clarity seems to be a crucial aspect of reliability not sufficiently covered by *reproducibility* and *replicability* as specified above. That said, *conceptual clarity* appears related to the concept of *validity* in quantitative research fields such as medicine or psychology, which describes "the extent to which a measure accurately represents the concepts it claims to measure" (Roberts & Priest, 2006).

In summary, we use the term *reliability* as an umbrella term that encompasses *reproducibility*, *replicability*, and *conceptual reliability* as specified above. While benchmark studies, the subject of Part II, can be a tool to improve reliability in supervised machine learning in terms of replicability, conceptual clarity appears particularly relevant for unsupervised learning, where a *ground truth* to compare results against is usually not available (Shalev-Shwartz & Ben-David, 2014; Zimmermann, 2020). Part III is devoted to reliability in unsupervised learning from a manifold learning perspective. We provide further background on these two aspects in the next chapters. Chapter 3 focuses on supervised benchmark studies and Chapter 4 on manifold learning and reliability in unsupervised learning.

3. Benchmark Studies and Reliability in Supervised Learning

> It is easy to obtain confirmations, or verifications, for nearly every theory — if we look for confirmations.
> — Karl Popper

3.1. Overview

Part II is concerned with supervised benchmark studies on real data. As already noted, Chapter 5 presents an example of such a study with a focus on survival prediction in cancer data. Comparing methods proposed elsewhere based on systematically selected real data sets, benchmark studies are an important tool to improve reliability in machine learning. However, researchers conducting benchmark studies face a multiplicity of design and analysis options, which can in turn affect the outcome of such studies. Chapter 6 provides a follow-up on the study presented in Chapter 5 elaborating on this issue.
That is, Part II is concerned with how to properly conduct *method comparisons* in supervised learning on multiple real data sets and in what follows we provide some background. Note that a substantial amount of work has been devoted to the question of how to draw conclusions about methods' performance differences based on real data, in particular, by *null hypothesis significance testing* (NHST). For example, Dietterich (1998), Alpaydin (1999), Nadeau & Bengio (2003), Bouckaert (2004), Bouckaert & Frank (2004), Bengio & Grandvalet (2004), and Hothorn et al. (2005) elaborate on performance comparison given a single data set, while Demšar (2006), Eugster et al. (2012), Boulesteix et al. (2015), and Eisinga et al. (2017) consider settings with multiple data sets. The latter is specifically relevant for methodological research as it is usually intended to generalize performance differences to other domains, i.e., to other than the observed real data sets (Boulesteix et al., 2015), and multiple data set comparisons have long been a practically established tool in machine learning research (Demšar, 2006). However, in contrast to the *single data setting*, only small advances have been achieved towards a sound statistical underpinning of method comparisons based on multiple real data sets (Boulesteix et al., 2015). It may be for that reason that the conducted experiments often do not justify the intended generalizations in practice. In particular, there are regularly too few data sets and, additionally, the data set selection is often considerably biased (Boulesteix et al., 2013). In the following, it is described why *comparison studies* based on a sufficient number of real data sets are crucial when comparing prediction methods and we exemplify why such comparisons should be conducted as *neutral* comparison studies.
Before going into the details, we briefly recap the principles of supervised learning. Note that with the main focus on *method comparison*, the following exposition does not cover all aspects of supervised learning in full detail. For example, we consider resampling approaches such as cross-validation (CV) the standard procedure for *model assessment* and *selection*, although there are other approaches such as *Structural Risk Minimization*, *AIC*, *BIC*, or *Minimum Description Length* (e.g., see Hastie et al., 2009, Ch. 7). Moreover, we do not explicitly elaborate on

important concepts such as the *Bias-Complexity-Tradeoff* (e.g., see Shalev-Shwartz & Ben-David, 2014, Ch. 5) and the difference between *parameters* and *hyperparameters* (e.g., see Guyon et al., 2010), assuming readers are familiar with the problem of *overfitting* and the importance of *hyperparameters* in controlling the *Bias-Complexity-Tradeoff*.

3.2. Principles of Supervised Learning

Learning Prediction Models via Empricial Risk Minimization In essence, the fundamental goal in supervised machine learning is to obtain (*learn*) a function or *prediction model* f that allows computing some unobserved output y from some observed input x known or assumed to be associated with y (Shalev-Shwartz & Ben-David, 2014). For example, one might be interested in the expected survival time of patients suffering from cancer given a set of features such as the patients' age, vital status, or genetic attributes, as is the case in the study presented in Chapter 5. Statistical learning theory now provides a regime to obtain f based on a finite sample $S = \{(x_1, y_1), ..., (x_n, y_n)\} \in X \times Y$ of initially observed input-output pairs (x_i, y_i), in such a way that it encodes (*learns*) also some information about the structure of the unknown joint distribution $P(X, Y)$ of which the input-output pairs are assumed to be independent and identically distributed (i.i.d.) random samples of (Hastie et al., 2009). This means, the prediction model f generalizes to some extent to i.i.d., but unseen data and thus allows to output y from inputs x that are not in the initially observed set S. Theoretically, one would like f to minimize the generalization error $\epsilon(f) = E[L(f(X), Y)]$, with E the expectation and L a loss function measuring the discrepancy between the actual outcome y and the predicted value $\hat{y} = f(x)$. But since $P(X, Y)$ is unknown, this is not feasible and it is approximated based on the observed data S by minimization of the empirical risk (Hastie et al., 2009; Shalev-Shwartz & Ben-David, 2014)

$$R_S(f) = \frac{1}{n} \sum_{i=1}^{n} L(f(x_i), y_i)). \tag{3.1}$$

Performance Assessment and Model Selection The basic criterion to assess a trained prediction model f is its prediction performance, which means how well it generalizes to unseen data. For that, an independent test set is required that has not been used in any way in training the model since the empirical risk $R_S(f)$ computed on instances used for training would be a biased estimate of the generalization performance. In the simplest case (the number of observations in S is large), S is split into non-overlapping training and test sets. The training set is used to fit the model via *empirical risk minimization* (ERM) and the test set is used to approximate the generalization error as a more accurate estimate of the prediction performance, i.e., for *performance assessment* (Hastie et al., 2009).
The *prediction model* f is usually selected from a set of candidate functions, often called *hypothesis space*, and the hypothesis space is usually restricted to a set of candidates from a specific *model class* that imposes a more or less specific structural assumption about possible prediction models f (this is called *inductive bias*) (Shalev-Shwartz & Ben-David, 2014). Performance assessment is thus closely related to *model selection* (Guyon et al., 2010) and selecting a model from the model class requires a further data split into train and validation set. *Model selection* then amounts to fitting different models f to the training data, computing the models' performance on the validation set, and selecting the one with the best prediction performance on the validation set. Again, the overall selected model's prediction performance

can then be assessed using the test set (Hastie et al., 2009). In general, this data splitting principle for model selection and performance assessment involves more complex resampling schemes such as repeated CV as conducted in the benchmark study presented in Chapter 5, particularly to account for small sample sizes. It should be noted that ERM induces an estimation error depending on the sample size n. Since CV reduces the training set size, it increases the estimation bias. For a detailed discussion of resampling approaches see, for example, Bischl et al. (2012).

To keep it simple, a model class is considered to be defined by a specific *prediction method* or *learning algorithm*, for example, Lasso regression or random forest. Such prediction *methods* can be adapted to specific tasks by a set of *hyperparameters* (Hastie et al., 2009) and performance assessment and model selection can be delineated based on the difference of (a method's) *hyperparameters*, which are specified during model selection, and (a model's) *parameters*, which are specified during ERM. Guyon et al. (2010) refer to the latter as the first level of inference and the former as the second level of inference. That said, it should be noted that *hyperparameter optimization (HPO)* can have a very general scope including preprocessing and postprocessing steps leading to complex machine learning pipelines to optimize over. There are supervised learning frameworks incorporating HPO in full generality (e.g., see Bischl et al., 2021), which go beyond the ones presented in standard references such as Hastie et al. (2009) or Shalev-Shwartz & Ben-David (2014).

3.3. Comparing Prediction Methods

With these principles in place, we return to the problem of comparing the performance of prediction methods. Choosing between different prediction methods is one of the fundamental goals in machine learning (Dietterich, 1998; see also Boulesteix et al., 2015) and doing so over different domains, i.e., based on different real data sets, "is perhaps the most fundamental and difficult question in machine learning" (Dietterich, 1998, p. 4). *Model selection* involves the comparison of different *models* on the validation set as well, but there is a crucial difference between performance assessment and model selection as specified above on the one side, and the identification of superior methods on the other side (Hothorn et al., 2005). Recall that (methodological) researchers usually intend to compare prediction *methods* in terms of their capabilities to produce prediction *models*. Moreover, they usually do not restrict their conclusions to the specific data sets at hand but (intend to) generalize the performance to other data sets not part of the data set selection used for the experiments (Boulesteix et al., 2015). Both differentiations are important and we discuss these two aspects based on the statistical framework provided by Boulesteix et al. (2015).

Comparing Methods versus Comparing Models Again, let P be the joint distribution over $X \times Y$ and let S denote an observed i.i.d. sample of size n drawn from P. Moreover, M denotes a *prediction method* and f_M^S a *prediction model* obtained by training M on S as outlined above. Then, Boulesteix et al. (2015) specify the generalization error as defined in Eq. 3.1 more precisely as the *conditional* generalization error

$$\epsilon(f_M^S, P) = E_P[L(f_M^S(X), Y)]. \tag{3.2}$$

Note that $\epsilon(f_M^S, P)$ is conditional on the method M, distribution P, and – in particular – to the observed sample S through the fitted model f_M^S (see also Hastie et al., 2009, Ch. 7). If the

goal is, for example, to compare two trained prediction models $f^S_{M_1}$ and $f^S_{M_2}$, approximating and comparing the conditional errors $\epsilon(f^S_{M_k}, P)$, $k = 1, 2$, is of major concern. Note that this is different from model selection as specified above as we compare models obtained with two different prediction methods M_1 and M_2. This sort of comparison is of more interest for applied researchers, who are given a specific data set they intend to analyze (Boulesteix et al., 2015) but also Hastie et al. (2009, Ch. 7) consider estimating the conditional error $\epsilon(f^S_M, P)$ as the major goal. In contrast, the *unconditional* generalization error is defined as

$$\epsilon(n, M, P) = E_{P^n}[\epsilon(f^{\mathscr{S}}_M, P)], \tag{3.3}$$

with $\mathscr{S}$ a random i.i.d. sample following the distribution P^n. The error $\epsilon(n, M, P)$ only depends on M, the sample size n, and the distribution P but is no longer conditional on the specific sample S. It is the expected value of the performance of a method M over different samples $\mathscr{S} \sim P^n$. It thus allows comparing the performance of two prediction methods M_1 and M_2 conditional on the underlying distribution P, in contrast to the comparison of two prediction models trained on a specific sample from that P (Boulesteix et al., 2015). Thus, comparing unconditional errors $\epsilon(n, M_1, P)$ and $\epsilon(n, M_2, P)$ is of more concern to methodological research as understood here because the intention is to compare *prediction methods* (Boulesteix et al., 2015). More precisely, to test for a significant difference in performance the test hypothesis can be defined as

$$\begin{aligned} & H_0 : \epsilon(n, M_2, P) - \epsilon(n, M_1, P) \geq 0 \\ \text{versus} \quad & H_1 : \epsilon(n, M_2, P) - \epsilon(n, M_1, P) < 0. \end{aligned} \tag{3.4}$$

However, while $\epsilon(n, M_k, P)$ can be approximated using resampling procedures such as CV or bootstrapping, it is not as straightforward to estimate the variance of such estimators, which is crucial for NHST (Nadeau & Bengio, 2003; see also Bates et al., 2021). For a specific form of CV Nadeau & Bengio (2003) provide a sound variance estimate but there is no "unbiased estimator of the variance of K-fold cross-validation" in general (Bengio & Grandvalet, 2004, p. 1089). In contrast, Hothorn et al. (2005) define a general framework for method comparisons based on CV on bootstrap samples from the given data set ensuring independence so that statistical tests can be applied for performance comparison. Yet, overall there is still no established standard to test for performance differences of methods "based on a real dataset with unknown underlying distribution" (Boulesteix et al., 2015, p. 204). Even more importantly, $\epsilon(n, M_k, P)$ is conditional on a specific distribution P. A synonym for distribution is *data generating process* (DGP) (e.g., see Hothorn et al., 2005) and we use the two terms interchangeably in the following. Researchers, however, usually intend to generalize methods' performance differences obtained on real data sets to other settings, i.e., DGPs. This is a crucial difference as the test reflected by the hypotheses in 3.4 allows to generalize performance differences to samples following the underlying distribution P that generated the observed data but not to samples generated by a different DGP (see also Hothorn et al., 2005). Instead, for the latter form of generalization one needs to assume a "distribution of distributions" (Boulesteix et al., 2015, p. 209) to define a suitable hypothesis. This reflects a crucial paradigm change.

Comparing Methods on Multiple Real Data Sets To test for performance difference over multiple real data sets, Boulesteix et al. (2015) define the test hypotheses as

$$\begin{aligned} & H_0 : E(\epsilon(N, M_2, \Phi)) - E(\epsilon(N, M_1, \Phi)) \geq 0 \\ \text{versus} \quad & H_1 : E(\epsilon(N, M_2, \Phi)) - E(\epsilon(N, M_1, \Phi)) < 0, \end{aligned} \tag{3.5}$$

with E the expectation over $\Phi : \Omega \to \mathcal{V}$, $\mathcal{V}$ a set of distributions, and $N : \Omega \to \mathbb{N}$, random variables generating a distribution P and a sample size n, respectively. That means the unconditional error $\epsilon(n, M_k, P)$ of a method M_k is the realization of the random variable $\epsilon(N, M_k, \Phi)$. Given estimates of the unconditional errors obtained via resampling on a given data set $D \sim P^n$, the hypotheses in 3.5 can be reformulated as

$$\begin{aligned} & H_0 : E(\epsilon(N, M_2, \mathbf{D})) - E(\epsilon(N, M_1, \mathbf{D})) \geq 0 \\ \text{versus} \quad & H_1 : E(\epsilon(N, M_2, \mathbf{D})) - E(\epsilon(N, M_1, \mathbf{D})) < 0, \end{aligned} \tag{3.6}$$

with $\mathbf{D}$ a random variable generating data sets D_j and its distribution conditional on $\Phi = P_j$ and $N = n_j$ equal to $P_j^{n_j}$. This is more amenable as we can estimate $\epsilon(n, M_2, D_j) - \epsilon(n, M_1, D_j)$ given a set of $j = 1, ..., J$ observed data sets of size n_j, while the distributions P_j underlying the data sets are of course unobservable (Boulesteix et al., 2015). However, there are two assumptions here. First of all, it must hold that the bias introduced by resampling to compute $\epsilon(n, M_k, D_j)$ is equal for both methods. Boulesteix et al. (2015) provide adjusted hypotheses to account for a situation where this is not the case.
In contrast, the second assumption is much more crucial as one needs to assume that the data sets D_j are surrogates for i.i.d. samples "from the set of all possible distributions in the considered area of application" (Boulesteix et al., 2015, p. 205), which we simply refer to as *DGP population* or *population of DGPs* in the following. Arguably, this assumption does not hold in many comparison studies as there is seldom a clearly defined *DGP population* from which the data sets are randomly sampled nor in any other way clearly defined inclusion criteria. In particular, in studies proposing a new method there is often even a strong bias in favor of a new method because researchers "tend to overfit their new method to specific example datasets" (Boulesteix et al., 2015, p. 207). This issue is discussed in more detail in Section 3.4.
In summary, all approaches based on a single data set essentially only allow to generalize performance differences to data stemming from the same DGP that generated the initially observed data set. In contrast, multiple real data set comparisons as discussed above are an approach to generalizing performance differences to data that is generated by another DGP, or using the terminology of Dietterich (1998), to other *domains*. However, this framework rests on strong assumptions about the considered data sets which are hard to meet in practice. Conducting neutral comparison studies accounts for this complexity at least to some extent and, in particular, protects against unfair comparisons.

Neutral Comparison Studies According to Boulesteix et al. (2013), a neutral comparison study fulfills three criteria:

1. *Focus on comparison*: The primary research goal is to conduct a comparison of methods proposed elsewhere. A neutral comparison study does not propose a new method.

2. *Neutrality*: The authors of a neutral comparison study should be approximately equally familiar with all methods under comparison.
3. *Systematic study design*: Data set, method, and evaluation criteria selection should be based on strict inclusion criteria.

The first two criteria protect against introducing implicit biases and conducting unfair comparisons, for example, by indirectly optimizing a method to the considered data sets or misspecification of an unfamiliar method. Beyond that, a systematic study design protects against overoptimistic findings and the data set selection plays a particularly important role. First of all, when it comes to NHST, power considerations are a crucial issue. For example, Boulesteix et al. (2015) provide an approach for a one-sided, one-sample t-test and emphasize that two sorts of variability must be accounted for. First, the *variability across data sets* is usually large and this variability can only be accounted for if enough data sets are included. Secondly, on each of the data sets the unconditional error is estimated for the methods under comparison, with all the *variability of error estimation* this entails. While this can be controlled to some extent by choosing appropriate resampling procedures, some effect of the data set sizes on the number of data sets needed remains (Boulesteix et al., 2015). Moreover, clearly defining and reporting data set inclusion criteria is crucial to avoid including data sets specifically fitting specific methods or dismissing data sets post hoc. Ideally, one randomly samples data sets from the domain of interest such that they yield enough power for NHST. However, the more narrow and specific the *population of DGPs* is defined and the more strict the inclusion criteria are, the less likely it will be to find available data sets meeting the requirements and ending up in the study. Consequently, there is often a trade-off between a well-defined domain on the one side and a sufficient amount of data sets on the other side. Finally, neutral comparison studies are usually not limited to two methods (as considered so far) but include several methods to compare. This requires suitable tests and possibly corrections for multiple testing.
In summary, systematically conducting a neutral comparison study includes specifying the domain of interest, deriving strict inclusion criteria, power calculations, and selecting data sets accordingly. Note that Boulesteix et al. (2017) compare neutral comparison studies to medical trials with methods playing the role of treatments and data sets the role of patients.

3.4. Implications for Reliability in Supervised Learning

The terms benchmark *study* and benchmark *experiment* are often used interchangeably in the literature, including the study presented in Chapter 5. For the following discussion it is convenient to distinguish these two terms. Hothorn et al. (2005) use the term *benchmark experiment* specifically for method comparisons based on a single data set and we follow this example. In contrast, the term *benchmark study* is used as a synonym for a *neutral comparison study*, i.e., studies based on multiple data sets that fulfill the three criteria outlined above.

Benchmark Studies versus Method Demonstrations Consequently, benchmark studies can be seen as a tool to come to (more reliable) conclusions about prediction methods' performance differences. In that, they have to be distinguished from method comparisons which are usually conducted as a part of a paper introducing a new method. Boulesteix (2013) calls such experiments *illustrative* method comparisons. In the following, they are referred to as *method demonstration* to more clearly distinguish them from benchmark studies because the main purpose there is to *demonstrate* the practical value of the new method. Contrasting it to other,

already existing methods using benchmark experiments is an important aspect. However, in contrast to a benchmark study, method demonstrations are far less suited to draw conclusions about the superiority of one method over the other. First of all, method demonstrations are usually based on a very limited number of data sets. For example, Boulesteix et al. (2013) report that often only up to 10 data sets are included. That means, method demonstrations are usually underpowered and do not allow drawing statistically significant conclusions alone for that reason. Moreover, they usually lack a systematic data set selection. Properly defining a *DGP population* to sample from is a general and fundamental problem also affecting benchmark studies but the data sets in method demonstration usually do not follow strict inclusion criteria and are often considerably biased in favor of the newly proposed method in addition. As Boulesteix et al. (2015, p. 207) emphasize, researchers

> "tend to overfit their new method to specific example datasets while developing them. The variance across datasets being high, this new method that has been optimized to these particular datasets is likely to perform much worse on other datasets."

This is all the more relevant as there is always a setting in which a method performs poorly while other methods perform well (Shalev-Shwartz & Ben-David, 2014, Ch. 5.1). Consequently, one of the driving forces for replicability issues in machine learning research appears to be that results of method demonstrations are used as (empirical) evidence for conclusions concerning the superiority of the newly proposed method over existing ones that generalize to other domains like the ones considered in the study (see also *SotA-hacking*, Gencoglu et al., 2019). This is not to say that method demonstrations are not useful and valuable contributions. This is more to emphasize the importance of benchmark studies and that method demonstrations are used for a purpose they are not suited for (comparing methods over distributions of DGPs), and not brought to their full potential for the purpose they are suited for (demonstrating a method's capabilities). That said, great effort is usually put into demonstrating where a method performs well. It would be of equally great value if a similar effort were made to also show where a method does not work or, as Rendsburg et al. (2020, p. 9) put it: "Finding examples where an algorithm works is important — but maybe even more important is to understand under which circumstances the algorithm produces misleading results". Recalling Feynman (1974), method demonstrations can thus be considered a perfect tool to illustrate "everything that you think might make it invalid—not only what you think is right about it" (of course, the pressure to report positive findings works against this). As will be outlined in Chapter 4, this is specifically relevant in unsupervised settings where approaches toward benchmark studies are still in their infancy.

Limitations of Benchmark Studies Despite their clear advantages over *method demonstrations* in terms of generalizability and their importance to improve reliability in machine learning research, there are also some important limitations of *benchmark studies*. First of all, the various design decisions a researcher faces when conducting and analyzing a benchmark study can affect the results. That means, benchmark studies also carry the risk of drawing improper and overly optimistic conclusions (Dehghani et al., 2021; cf. Li et al., 2019). The study presented in Chapter 6 illustrates the effects of different data sets but also of different performance measures, aggregation methods, and approaches to handle missing performance values.

The most crucial aspect arguably remains the data set selection and we, therefore, discuss it in more detail here. The fundamental problem is that without being able to come up with a precise definition of the population of DGPs, it can – from a very principled perspective – be

argued that it is difficult to define what a benchmark study measures. In other words, even if the selected data sets are randomly sampled and provide enough power to detect significant differences, it is not clear what this generalizes to.
First of all, the domain of interest is often simply defined to be data sets from a specific database. This is certainly a useful and practically feasible approach that allows specification of further inclusion criteria such as the number or type of features or some other external characteristics. But in principle this does not solve the issue as it is questionable that a sample of data sets from a given database (even if randomly drawn) generalizes to data sets from other databases. Boulesteix et al. (2015, p. 205) emphasize this issue by stating that "it may be difficult, if not impossible, to draw independent realizations from the set of all possible distributions in the considered area of applications". More generally, recall the statement of Hand (2006, p. 12) (who they also cite) that the considered data set "collection will not be representative of real data sets in any formal sense". Moreover, note that the choice of a prediction method "should ideally be based on some prior knowledge about the problem to be learned" (Shalev-Shwartz & Ben-David, 2014, p. 37) as this choice introduces *inductive bias* (before any training takes place). In addition, another important aspect is that there is no guarantee that the considered data sets are intrinsically similar to each other in any way whatsoever. For example, two data sets that are very different in terms of numbers of features or any other external characteristic can be very similar to each other in terms of intrinsic properties such as the extent of class separability or the number of distinct clusters.
It appears to be one of the most pressing problems to come up with approaches that reflect all these aspects and which allow to consistently define common characteristics of the data sets, i.e., to more clearly define common structures and characteristics of relevant DGPs in the domain of interest.

A Note on Reproducibility In summary, benchmark studies are an important contribution toward more reliable machine learning in terms of replicability. Yet, they may not be easily reproducible. In general, Li et al. (2019) describe several aspects affecting the reproducibility of benchmark studies including issues arising from hardware and programming language differences. Another important aspect is that benchmark studies that take several competing methods into account based on a sufficient amount of data sets to allow, for example, for significance testing, quickly require considerable computation times and resources. For example, the benchmark study comparing 13 methods on 18 data sets presented in Chapter 5 took about two weeks to compute in parallel on a personal computer. While this appears still feasible, the computations of a study comparing time series classification methods on 85 data sets took more than six months on a High-Performance-Cluster (HPC) (Bagnall et al., 2017), which is less likely to be reproduced. Similar holds for state-of-the-art deep learning approaches, for example, in natural language processing (Aßenmacher et al., 2021). Considering these limiting factors, it may be better to put the effort into replicating a benchmark study on different data (for example, with data from a different database) instead of exactly reproducing it because this implicitly broadens insight into the domains the method comparison is valid on. Moreover, this again stresses the importance of reproducibly demonstrating a method's capacities and caveats already on the level of papers introducing new methods, as those are usually limited to a smaller number of data sets and competing methods and thus are more likely to be reproduced.

4. Manifold Learning and Reliability in Unsupervised Learning

There is no unique picture of reality. — Stephen Hawking

4.1. Overview

Manifold learning is based on the assumption that observed high-dimensional data only occur on or near to a typically non-linear, lower-dimensional manifold embedded in the observation space. That means that the data is intrinsically actually much lower dimensional than observed. Manifold learning methods try to (1) infer this intrinsic structure and (2) find a low-, usually 2-dimensional representation that faithfully preserves the crucial characteristics of the manifold (Lee & Verleysen, 2007; Ma & Fu, 2011). The manifold assumption is usually formalized as follows: the high-dimensional data observed in a D-dimensional space $\mathcal{H}$ lie on or close to a d-dimensional manifold $\mathcal{M} \subset \mathcal{H}$, with $d < D$ (Cayton, 2005). The goal is to infer an embedding function $e : \mathcal{H} \to \mathcal{Y}$ from the high-dimensional space to a low-dimensional embedding space $\mathcal{Y}$ such that $\mathcal{Y}$ is as similar to $\mathcal{M}$ as possible. Often it is simply considered that $\mathcal{H} = \mathbb{R}^D$. Note that the terms manifold learning and nonlinear dimensionality reduction are used interchangeably in the literature (Cayton, 2005; Gisbrecht & Hammer, 2015; Lee & Verleysen, 2007; Ma & Fu, 2011).

Many methods have been developed (or can be used) for manifold learning. For example, this includes multidimensional scaling (MDS, Cox & Cox, 2008), generalized principal component analysis (GPCA, Vidal et al., 2016), Isomap (Tenenbaum et al., 2000), Local Linear Embeddings (LLE, Roweis & Saul, 2000), Laplacian Eigenmaps (Belkin & Niyogi, 2003), Diffusion Maps (Coifman & Lafon, 2006), t-distributed stochastic neighbor embedding (t-SNE, Maaten & Hinton, 2008), and uniform manifold approximation and projection (UMAP, McInnes et al., 2020) to name only a few prominent examples. Most of these methods share a fundamental design principle consisting of two steps: (1) constructing a weighted k-nearest-neighbor (k-NN) graph from a pairwise distance matrix and (2) finding a (low-dimensional) representation of the graph which preserves as much of its structure as possible. The methods differ in how exactly they perform these two steps. However, most methods crucially depend on the hyperparameter steering the neighborhood size. We describe MDS and UMAP, the two methods of particular relevance in Part III, in more detail in Section 4.3.

Manifold learning as considered here has found widespread application in many domains, including single cell data (e.g., see Becht et al., 2019; Kobak & Berens, 2019), multi-omics cancer data (Cantini et al., 2021), or cardiac arrhythmia classification (Rajagopal & Ranganathan, 2017), to name only a few examples. However, it should be noted that this constitutes a selected perspective on manifold learning, that there are other approaches based on different principles, for example, neural-network-based approaches such as self-organizing maps and autoencoders, and that manifold learning can be regarded as a part of representation learning (Agrawal et al., 2021; Bengio et al., 2013; Gisbrecht & Hammer, 2015; Lee & Verleysen, 2007).

4.2. Problem Specification

The most prominent approach to structure manifold learning is to differentiate methods according to whether they focus on *local* or *global* characteristics (e.g., see Becht et al., 2019; Cayton, 2005; Y. Wang et al., 2021). *Local* methods focus on preserving the local neighborhood structures of the observations, while *global* methods "aim mainly to preserve relative distances or rank information between points, resulting in a stronger focus on the preservation of distances between points that are farther away from each other" (Y. Wang et al., 2021, p. 4). However, terminology and concepts are not consistent overall. First of all, note that Cayton (2005, p. 13) and Ma & Fu (2011, p. 2) consider Isomap a global method because it tries to preserve all pairwise (geodesic) distances. Y. Wang et al. (2021, p. 5f), in contrast, consider Isomap – together with LLE and Laplacian Eigenmaps – a local method because it tries "to preserve local Euclidean distances from the original space when creating embeddings". In addition, Lee & Verleysen (2007, Ch. 7.4) provide a taxonomy differentiating *topology preserving* and *distance preserving* methods, where Isomap belongs to the former and LLE to the latter, for example. The difference is that *distance preserving* methods operate on a matrix of all pairwise distances (cf. *global*, Cayton, 2005; Ma & Fu, 2011), in contrast to *topology preserving* methods which operate on sparse distance matrices.
Finally, note that with regard to cluster analysis, some sources argue that "the existence of one or several underlying manifolds must be questioned" and that the "manifold assumption is probably wrong (or useless)" (Lee & Verleysen, 2007, p. 242). This understanding of the manifold assumption is in stark contrast to the following aspects: first of all, modern manifold learning methods, in particular t-SNE and UMAP, are used to draw conclusions about cluster structure in practice (Becht et al., 2019; Kobak & Berens, 2019). Moreover, it has been shown – experimentally for UMAP (Allaoui et al., 2020), theoretically for t-SNE (Linderman & Steinerberger, 2019) – that they can be used for clustering. Third, clustering can be considered a natural example of topological data analysis (Niyogi et al., 2011) and, as outlined, many manifold learning methods are considered *topology preserving*.
Moreover, the manifold learning methods considered above do not explicitly learn the mapping e but provide only vector representations of the high-dimensional observations in the low-dimensional embedding space. This has two important consequences: first of all, it is not straightforward to embed new data points into an existing embedding without recomputing the embedding completely. More importantly, it is not possible to assess the quality of an embedding, i.e., a method's performance, using the reconstruction error $E[L(x, e^{-1}(y))]$, with L a loss function quantifying the difference between $x \in \mathcal{H}$ and its reconstruction $e^{-1}(y)$, $y \in \mathcal{Y}$ (Lee & Verleysen, 2009). The reconstruction error would reflect a well-defined and objective criterion for performance assessment and method comparison. We discuss alternative performance measures in Section 4.3.3.
In general, this raises the question of what these different, underlying conceptual perspectives on manifold learning actually mean and how to assess them properly. In particular, the notion of *local* and *global* is usually specified by reference to differences in methods, without precisely specifying what exactly is meant by *local* and *global* from a problem- or data-driven perspective, i.e., whether *global* and *local* means the same in all data set and analysis situations. For example, does it mean the same in a situation where outliers are present or where the data is distributed in clusters? Moreover, is the assumption of a single manifold appropriate for such settings? Note that Lee & Verleysen (2007, Ch. 7.7) consider a setting with disconnected manifolds a general open question and, in particular, different from a setting with cluster structure. Lacking a well-defined and objective criterion to assess manifold learning outputs, this vagueness can easily lead to overoptimistic, misleading, or wrong conclusions about

methodological aspects as well as domain-specific aspects (e.g., see Kobak & Linderman, 2021; Y. Wang et al., 2021; Wattenberg et al., 2016). We demonstrate in Part III that the standard manifold assumption of a single (connected) manifold does not sufficiently reflect outlier and cluster structure. To overcome the issue, we argue that it is important to more clearly differentiate between the *inner* geometry, the *outer* geometry, and the *topology* of a data set (cf. Lee & Verleysen, 2007, Ch. 7.4; Tenenbaum et al., 2000). We give an introductory overview of these concepts in Section 4.5.3.

4.3. Methods

The methods MDS, Isomap, Diffusion Maps, t-SNE, and UMAP are applied in Part III. Of particular interest, however, are MDS and UMAP. The focus lies on MDS in Chapters 8 and 9, while Chapter 10 focuses on UMAP. We briefly discuss these two methods in the following.

4.3.1. MDS

Given (non-negative) dissimilarities between a set of n objects, the general aim of MDS is to find a layout or map, more precisely a set of coordinates usually in Euclidean space, where a point reflects an object and distances between the n points in the map should reflect the observed pairwise dissimilarities as closely as possible. Being originally developed as a tool in psychometrics and not for dimensionality reduction, these dissimilarities may also reflect, for example, interpersonal preferences that a psychologist assigns to a group of people. That said, MDS can be considered a method class rather than a single method depending on the dissimilarities used. That is also reflected by the fact that usually (classical) metric MDS is differentiated from non-metric or ordinal MDS (Cox & Cox, 2008; Mead, 1992). The major difference lies in the properties the dissimilarities have to fulfill.

Definition 4.1 (Metric space)**.** Let $\mathcal{X}$ be a set. A *metric* is a mapping $d_m : \mathcal{X} \times \mathcal{X} \to \mathbb{R}, (x_i, x_j) \mapsto d_m(x_i, x_j)$ with the following properties:

1. $d_m(x_i, x_j) = 0$ iff $x_i = x_j$
2. $\forall x_i, x_j \in \mathcal{X} : d_m(x_i, x_j) = d_m(x_j, x_i)$ (Symmetry)
3. $\forall x_i, x_j, x_k \in \mathcal{X} : d_m(x_i, x_k) \leq d_m(x_i, x_j) + d_m(x_j, x_k)$ (Triangle inequality)

A metric space $\mathcal{X}_{d_m}$ is then a tuple $(\mathcal{X}, d_m)$.

Metric MDS requires the dissimilarities to fulfill all three metric properties, while non-metric MDS does not require the dissimilarities to adhere to the triangle inequality (Mead, 1992). In the following, we outline how to use metric MDS for dimensionality reduction, i.e., how to obtain a d-dimensional representation of a D-dimensional data set via metric MDS ($d < D$). For a detailed discussion of MDS in general see, for example, Mead (1992) or Cox & Cox (2008), and for more details from the manifold learning perspective, see Lee & Verleysen (2007, Ch. 4.2.2) or Ma & Fu (2011, Ch. 1.4.2).
Let $X \subset \mathbb{R}^D$ be a data set of n observations and $\Delta = (\delta_{ij})$ a $(n \times n)$-distance matrix with $\delta_{ij} = d_m(x_i, x_j), x_i, x_j \in X$. That is, X can be considered a subset of the metric space $(\mathbb{R}^D, d_m)$. Given Δ, lower dimensional representations $y_1, ..., y_n$ (*principal* coordinates) in a d-dimensional embedding space $\mathcal{Y} = \mathbb{R}^d$ can be computed via MDS as follows (see Cox & Cox, 2008, p. 319; Ma & Fu, 2011, p. 13):

1. Compute a matrix $A = (-\frac{1}{2}\delta_{ij}^2)$
2. Compute a matrix $B = HAH$, with $H = I - n^{-1}\mathbf{1}_n\mathbf{1}_n^T$ the centering matrix and $\mathbf{1}_n$ a vector of ones.
3. Compute $B = V\Lambda V^T$, the spectral decomposition of B, with $\Lambda = \text{diag}\{\lambda_1, \cdots, \lambda_n\}$ a diagonal matrix of the (decreasingly ordered) eigenvalues of B, and $V = (\mathbf{v}_1, ..., \mathbf{v}_n)$ the matrix of corresponding eigenvectors[1].
4. Compute $V_d\Lambda_d^{\frac{1}{2}} = (\sqrt{\lambda_1}\mathbf{v_1}, ..., \sqrt{\lambda_d}\mathbf{v_d}) = Y$. The rows of Y yield the d-dimensional coordinates $y_1, ..., y_n \in \mathcal{Y}$.

To assess the distortion induced by reducing the dimension to d, one can compute the *goodness of fit* of the embedding via (Cox & Cox, 2008)

$$\frac{\sum_{i=1}^{d}\lambda_i}{\sum_{i=1}^{n-1}|\lambda_i|} \quad \text{or} \quad \frac{\sum_{i=1}^{d}\lambda_i}{\sum_{j\in\{i:\lambda_i>0\}}\lambda_j}.$$

In most applications of manifold learning, Δ is computed using simple Euclidean (L_2) distances $d_{euc}(x_i, x_j) = \sqrt{\sum_{l=1}^{D}(x_{il} - x_{jl})^2}$ between the observations in the high-dimensional observation space $\mathcal{H}$, for which lower dimensional representations in $\mathbb{R}^d$, $d < D$, are sought (Ma & Fu, 2011, Ch. 1). In this case, MDS is equivalent to PCA. If, however, another metric is chosen, for example, a different L_p metric, this no longer holds (Cox & Cox, 2008). For that reason we consider PCA a special form of MDS.

Similarly, we consider Isomap a special form of MDS (see also Lee & Verleysen, 2007 Ch. 4.3.2; Ma & Fu, 2011 Ch. 1.5.1). Given a distance matrix Δ (usually, but not necessarily, of L_2 distances) of high-dimensional observations as input, Isomap consists of three steps (Tenenbaum et al., 2000):

1. Compute a k-NN graph G with edge weight δ_{ij} if x_i is in the k-neighborhood of x_j.
2. Given G, compute the shortest path distances between all pairs of points.
3. Use the resulting $n \times n$ matrix Δ_{geo} of shortest path distances as input to MDS.

In other words, Isomap is MDS applied to a distance matrix of shortest path distances. These distances approximate the *geodesic* distances on the nonlinear lower-dimensional manifold $\mathcal{M} \subset \mathcal{H}$ the observed high-dimensional data are assumed to lie on (or near to). The important difference is that MDS based on geodesic distances preserve all pairwise distances according to the manifold, i.e., the intrinsic or *inner* geometry of the data manifold, while MDS based on Euclidean distances preserves the *outer* geometry, i.e., the geometry inherited from the *ambient* space (Tenenbaum et al., 2000).

Using *geodesic* distance (based on a properly chosen k) results in coordinates $y_1, ..., y_n$ such that a 2-dimensional, unclosed surface non-linearly embedded in $\mathbb{R}^3$ is unfolded in a 2-dimensional embedding with little or no distortion at all. In contrast, using L_2 distances instead, a 2-dimensional embedding will lead to distortions because the manifold is essentially projected on the linear subspace spanned by the eigenvectors corresponding to the largest variance. This is the reason why MDS is considered a linear method. However, if Δ is a L_2 distance matrix, the inner product matrix B is positive semi-definite. Computing an embedding of dimension $d = \text{rank}(B)$ with MDS based on Δ will lead to coordinates $y_1, ..., y_n$ with

[1] Given distances based on the Euclidean metric, B is non-negative definite. If $\text{rank}(B) = t < n$, the largest t eigenvalues are positive and the remaining 0. If distances are computed with another metric, negative eigenvalues can occur.

Euclidean distance matrix exactly matching Δ (Young & Householder, 1938; see also Cox & Cox, 2008; Torgerson, 1952). This means, if the D-dimensional data live on a nonlinear manifold in a d-dimensional subspace, constructing a d-dimensional embedding with MDS based on L_2 distances will exactly reproduce the manifold. This is important to keep in mind for the functional data setting. There it is straightforward to produce, say, a 50-dimensional observation space based on a 2-dimensional nonlinear manifold that can then be reconstructed in terms of its *outer* geometry using MDS (see Chapter 7). Similarly, assuming a data set to consist of two disconnected manifolds, it does not make sense to compute geodesic distances between all observations, i.e., inferring the *inner* geometry, but it does make sense to compute L_2 distances, i.e., inferring the *outer* geometry, a fact relevant for outlier detection as outlined in Chapters 8 and 9. In summary, the mapping $e_{MDS} : \mathcal{H} \to \mathcal{Y}$ implicitly learned by MDS is an *isometric* mapping and MDS tries to preserve the metric structure induced by the metric d_m chosen to compute the pairwise distances between observations.

4.3.2. UMAP

The method UMAP is based on three important assumptions (or "axioms"). It is assumed that "there is a manifold on which the data would be uniformly distributed", "the underlying manifold of interest is locally connected", and "preserving the topological structure [...] is the primary goal" (McInnes et al., 2020, p. 13). In its basic computational structure, UMAP is rather similar to MDS and other manifold learning methods and we briefly outline the computational side of UMAP in the following. We then try to build up some intuition about the theoretical underpinnings of the computational steps.
Given a distance matrix Δ (of observations in a high-dimensional data set X), UMAP computes an embedding by first constructing a weighted k-nearest-neighbor graph G (graph construction step) and then finding a representation based on G (graph layout step). In particular, first a *directed* graph $\bar{G}$ is computed, where edge weights are defined by the weight function

$$v((x_i, x_{i_j})) = \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right), \tag{4.1}$$

with x_{i_j} a k-nearest-neighbor of x_i, ρ_i the distance to the nearest neighbor of x_i, and σ_i a normalization factor specific to x_i (McInnes et al., 2020). This means that there are different (local) metric spaces around each x_i due to ρ_i and σ_i. This is necessary as the underlying theory requires the data to be uniformly distributed on the manifold, which is not a realistic assumption for real-world data. Defining different metrics at each point allows circumventing this issue. Moreover, note that ρ_i in Eq. 4.1 implies that x_i is at least connected to its nearest neighbor. This *local connectivity constraint* warrants that a single observation is not completely separated from the rest of the observations. The weighted adjacency matrix A of the directed graph $\bar{G}$ can then be transformed into an adjacency matrix $B = A + A^T - A \circ A^T$, with $\circ$ the pointwise product, of an undirected graph G. The resulting edge weights of G can be understood as the probability that the edge exists. Low-dimensional representations $y_1, ..., y_n$ are then obtained by minimizing the cross entropy

$$C_{UMAP} = \sum_{i \neq j} v_{ij} \log\left(\frac{v_{ij}}{w_{ij}}\right) + (1 - v_{ij}) \log\left(\frac{1 - v_{ij}}{1 - w_{ij}}\right) \tag{4.2}$$

via stochastic gradient descent (SGD), with v_{ij} representing the (dis)similarity based on G and $w_{ij} = (1 + a||y_i - y_j||_2^{2b})^{-1}$. Importantly, the two parts in Eq. 4.2 reflect attractive and repulsive forces (McInnes et al., 2020). That means, C_{UMAP} becomes minimal if $v_{ij} = w_{ij}$. If $v_{ij} = 0$, the embedding vectors y_i and y_j should be placed as far from each other as possible as this will drive w_{ij} towards zero. In particular, this means UMAP increases inter-cluster distances and decreases intra-cluster distances (see Chapter 10).
From a theoretical perspective, UMAP builds on sophisticated topological underpinnings and is closely related to topological data analysis (TDA). Note that we can not describe the theoretical aspects underlying UMAP and TDA in full detail but instead we try to build up some intuition. In particular, the basic conceptual and computational building blocks of both approaches are *simplicial complexes*. Intuitively, a *simplicial complex* K can be seen as the set of $\mathbf{k}$-*simplices* that can be constructed based on the $\mathbf{k}+1$ points in a given set. That means, points are 0-*simplices*, edges between points are 1-*simplices*, triangles are 2-*simplices* and so on. Moreover, a *face* of a $\mathbf{k}$-*simplex* is a subset and thus a (lower order) *simplex* itself. Each $\mathbf{k}$-*simplex* consists of $\mathbf{k}+1$ points (0-faces), and a triangle, for example, additionally has three edges (1-faces) (Wasserman, 2016; Zomorodian & Carlsson, 2005). Chazal & Michel (2021, p. 3f), for example, provide the following more precise definitions.

Definition 4.2 ($\mathbf{k}$-dimensional simplex)**.** Given a set $\mathbb{X} = \{x_0, ..., x_\mathbf{k}\} \subset \mathbb{R}^D$ of $\mathbf{k}+1$ affinely independent points, the $\mathbf{k}$-dimensional simplex $\mathbf{s} = [x_0, ..., x_\mathbf{k}]$ spanned by $\mathbb{X}$ is the convex hull of $\mathbb{X}$. The points of $\mathbb{X}$ are called the vertices of $\mathbf{s}$, and the simplices spanned by the subsets of $\mathbb{X}$ are called the faces of $\mathbf{s}$.

Definition 4.3 (Geometric simplicial complex)**.** A geometric simplicial complex K in $\mathbb{R}^D$ is a collection of simplices where any face of a simplex of K is a simplex of K and the intersection of any two simplices of K is either empty or a common face of both.

Definition 4.4 (Abstract simplicial complex)**.** Given a set $\mathcal{X}$, an abstract simplicial complex with the vertex set $\mathcal{X}$ is a set $\tilde{K}$ of finite subsets of $\mathcal{X}$ such that the elements of $\mathcal{X}$ belong to $\tilde{K}$ and for any $\mathbf{s} \in \tilde{K}$, any subset of $\mathbf{s}$ belongs to $\tilde{K}$.

Chazal & Michel (2021, p. 4) emphasize that "abstract simplicial complexes can be seen as topological spaces and geometric complexes can be seen as geometric realizations of their underlying combinatorial structure". This has two important implications from a data analysis perspective. First of all, constructing a simplicial complex from a given *data* set $X = \{x_1, ..., x_n\}$ allows to infer topological features of the data such as connected components (clusters) or holes. Secondly, since simplicial complexes are combinatorial objects they allow for efficient computations (Chazal & Michel, 2021).
There are different ways to obtain simplicial complexes from data in practice. For example, constructing Čech complexes $\check{C}_r(X)$ or Vietoris-Rips complexes $VR_r(X)$ are common approaches (Chazal & Michel, 2021; Wasserman, 2016; Zomorodian & Carlsson, 2005). Given a set of points of a metric space $\mathcal{X}_{d_m}$ and $r \in \mathbb{R}_0^+$, a Vietoris-Rips complex is the set of simplices $[x_0, ..., x_\mathbf{k}]$ with $d_m(x_i, x_j) \leq r$. In contrast, a Čech complex is the set of simplices such that the $\mathbf{k}+1$ closed balls $B(x_i, r)$ of radius r have a non-empty intersection (Chazal & Michel, 2021). The value r thereby steers the *resolution* at which the topological features are inferred from the (finite) data set. Basically, $r = 0$ will let each observation (point) appear as one of n unconnected components. That means, no observation is connected with another observation, respectively every connected component is a single data point. A very large r, on the other hand, will result in one single connected component, i.e., all observations will appear connected to each other (Bubenik, 2015).
A family of nested simplicial complexes $(K_r)_{r \in I}$ is called a *filtration*, when $K_r \subseteq K_{r'}$ if $r < r'$

for any pair $r, r' \in I \subseteq \mathbb{R}$ (Chazal & Michel, 2021). Computing filtrations, that is, a series of Čech or Vietoris-Rips complexes for increasing values of r, is a very important approach in TDA called *persistent homology* (Wasserman, 2016; see also Chazal & Michel, 2021; Zomorodian & Carlsson, 2005). Wasserman (2016, p. 17) states that "homology characterizes sets based on connected components and holes". That means, starting from the observations in a data set as n (un)connected components, other topological features, including holes or voids, will appear and vanish with increasing r (Wasserman, 2016). The *birth* and *death* times of these topological features can be used to create a *persistence* diagram, with the birth time plotted on the horizontal axis and the death time plotted on the vertical axis. *Persistent* topological features, i.e., features with long lifetimes, appear far from the diagonal. If, for example, a data set consists of three well separable clusters of similar density, the persistence diagram would indicate three persistent connected components. If a data set consists of observations drawn from a circle and uniformly distributed noise points, the persistence diagram would indicate a single persistent connected component and a single persistent hole. Note that there are approaches based on bootstrapping to assess the statistical significance of persistent topological features (Wasserman, 2016).

UMAP, on the other hand, constructs a single Vietoris-Rips complex in its graph construction step in principle. And if the data were uniformly distributed on the manifold, this would already yield a good approximation. However, since uniformity is too strong an assumption in reality, the Vietoris-Rips complex is not constructed based on a resolution or radius r but instead by specifying a number of nearest neighbors k to include. This means, the (individual) radius of the ball $B(x_i, r_k)$ around a point x_i is determined by the distance to its k-th neighbor (and not by a fixed r). Since k is the same for all points, this results in locally different metrics (smaller radii in dense regions, larger radii in sparse regions) (McInnes, 2018). In particular, it does not necessarily hold that $d_{m_i}(x_i, x_j) = d_{m_j}(x_j, x_i)$, i.e., each edge (1-simplex) can have two different weights depending on the point (0-face) from which the distance is measured. More specifically, this means one obtains a family of fuzzy simplicial sets. A difference between simplicial sets and simplicial complexes is that the former include directed edges, while the latter do not. Moreover, the simplicial sets are fuzzy because of the two different weights an edge can obtain and a fuzzy union of the fuzzy simplicial sets results in an undirected graph approximating the underlying manifold structure (McInnes, 2018).

In summary, that means that both TDA and UMAP approximate continuous structures by building simplicial complexes and sets from the (discretely) observed data points. UMAP constructs a topological representation of a manifold assumed to underlie the data using fuzzy simplicial sets and persistent homology, as a specific example of TDA, yields a persistence diagram that indicates (statistically significant) topological features based on filtrations. These close connections between TDA and UMAP emphasize UMAP's *topological* "nature". In particular, if the parameter k steering the neighborhood size is much smaller than the number of observations n, many edge weights will become 0 in the graph construction step. On the other hand, the local connectivity constraint ensures that the data set is not separated into many connected components consisting of a single point only. Together with optimizing the cross-entropy in the graph layout step this results in increasing inter-cluster distances and decreasing intra-cluster distances. In general, that means $e_{UMAP} : \mathcal{H} \to \mathcal{Y}$ should not be considered an isometric mapping but rather a *homeomorphism* (a function preserving topological features but not distances). Chapter 10 elaborates on this aspect in more detail and demonstrates how this can considerably enhance cluster analysis in practice. On the other hand, unlike MDS, UMAP appears not very suited for outlier detection as understood in Chapters 8 and 9, where it is important to *isometrically* retain the *outer* geometry.

4.3.3. Performance Measures

As already pointed out in Section 4.2, the reconstruction error $E[L(x, e^{-1}(y))]$ is often not available in manifold learning. Yet, there are alternative approaches to assess the quality of an embedding (e.g., see Kraemer et al., 2018; Lee & Verleysen, 2009). For example, just as in cluster analysis and outlier detection, external information such as labels can be used to assess an embedding (Lee & Verleysen, 2009). Another important approach is based on comparing g-neighborhoods in the high-dimensional space $\mathcal{H}$ and the low-dimensional embedding space $\mathcal{Y}$ specified by the ranks of pairwise distances, which is why they are commonly referred to as rank-based criteria (Lee & Verleysen, 2009). We also use the term *surrogate* performance measures to stress their heuristic basis. For example, this includes the trustworthiness and continuity (T&C) measures (Venna & Kaski, 2001), the local continuity meta-criterion (LCMC) (L. Chen & Buja, 2009), and the more recent local rank correlation (LRC) (Liang et al., 2020). A simple example of such a criterion is

$$Q_{NX}(g) = \frac{1}{g}\frac{1}{n}\sum_{i=1}^{n} |\mathcal{N}_g^{\mathcal{H}}(i) \cap \mathcal{N}_g^{\mathcal{Y}}(i)|, \tag{4.3}$$

where $\frac{1}{g}$ is a normalization factor and $\mathcal{N}_g^{\mathcal{H}}(i)$ and $\mathcal{N}_g^{\mathcal{Y}}(i)$ reflect the neighborhood of size g of observation i in the high-dimensional space $\mathcal{H}$ and the embedding space $\mathcal{Y}$, respectively (L. Chen & Buja, 2009). We use measures based on the LCMC criterion to automatically select hyperparameters of manifold learning methods in Chapter 7.
Note that Lee & Verleysen (2009) provide a unifying perspective on ranking-based criteria using the so-called *co-ranking* matrix (Lee & Verleysen, 2008). Following Lueks et al. (2011), the co-ranking matrix $\mathbf{Q}$ has entries

$$q_{kl} = |\{(i,j) : r_{ij}^{\mathcal{H}} = k \quad \text{and} \quad r_{ij}^{\mathcal{Y}} = l\}|, \tag{4.4}$$

with

$$r_{ij}^{\mathcal{H}} = |\{k : \delta_{ik} < \delta_{ij} \quad \text{or} \quad (\delta_{ik} = \delta_{ij} \quad \text{and} \quad 1 \leq k < j \leq n)\}| \quad \text{and}$$
$$r_{ij}^{\mathcal{Y}} = |\{k : d_m(y_i, y_k) < d_m(y_i, y_j) \quad \text{or} \quad (d_m(y_i, y_k) = d_m(y_i, y_j) \quad \text{and} \quad 1 \leq k < j \leq n)\}|$$

the neighborhood ranks in $\mathcal{H}$ (recall $\Delta = (\delta_{ij})$) and the embedding space $\mathcal{Y}$, respectively. For example, criterion 4.3 can then be reformulated as

$$Q_{NX}(g) = \frac{1}{g}\frac{1}{n}\sum_{i=1}^{g}\sum_{j=1}^{g} q_{ij}. \tag{4.5}$$

One can differentiate (mild and hard) *g-intrusions* and (mild and hard) *g-extrusions* according to the matrix $\mathbf{Q}$. Mild and hard *g-intrusion* are defined as $r_{ij}^{\mathcal{Y}} < r_{ij}^{\mathcal{H}} \leq g$ and $r_{ij}^{\mathcal{Y}} \leq g < r_{ij}^{\mathcal{H}}$, respectively (mild and hard *g-extrusion* are defined accordingly by changing the direction of the inequalities). Intrusions and extrusions are reflected in the lower and upper triangle of $\mathbf{Q}$, respectively. A perfect embedding would result in a diagonal co-ranking matrix (Lueks et al., 2011). Important here is that the LCMC criterion yields a single scalar value to measure mild intrusions and extrusions, which makes it well suited for hyperparameter tuning. On the

other hand, this means it does not differentiate between (mild) intrusion and extrusion. T&C, for example, differentiates between (hard) intrusion and (hard) extrusion and, respectively, expresses the effects on *trustworthiness* and *continuity* by two different values (Lee & Verleysen, 2009).

4.4. Manifold Learning in Functional Data

Functional data analysis (FDA) (Ferraty & Vieu, 2006; Ramsay & Silverman, 2005; J.-L. Wang et al., 2016) is an active research area in statistics. The focus lies on data where the units of observation are realizations of stochastic processes over compact domains. While functional PCA is a highly investigated topic (Happ et al., 2019; Happ & Greven, 2018; Shang, 2014) and there are approaches to infer manifold means and modes of variation (D. Chen & Müller, 2012) as well as template curve estimation (Dimeglio et al., 2014) on simple functional manifolds, manifold learning has not found widespread application in functional data analysis in general. From a manifold learning perspective, however, functional data has three important properties. Functional data

1. is high-dimensional but highly structured.
2. is theoretically/analytically well accessible.
3. can be easily visualized in bulk.

The first characteristic means that the manifold assumption is specifically realistic and useful. On the one hand, functional data is usually observed/measured at a large number of evaluation points resulting in high-dimensional data sets. On the other hand, there are usually only a few modes of variation, i.e., a low intrinsic dimensionality.
The second characteristic allows to precisely define and investigate different manifolds theoretically and, as a consequence, to simulate ostensibly complex, D-dimensional data based on precisely definable d-dimensional manifolds: Let $\phi : \Theta \to \mathcal{F}$ be a mapping from some parameter space $\Theta \subset \mathbb{R}^d$ to a function space $\mathcal{F}$. Usually, $\mathcal{F} = \mathcal{L}^2(\mathcal{T})$, the space of square integrable functions over domain $\mathcal{T}$. The parameter space Θ and the mapping ϕ specify a functional manifold $\mathcal{M}_\mathcal{F} \subset \mathcal{F}$ that can be defined as $\mathcal{M}_\mathcal{F} = \{x(t) : x(t) = \phi(\theta) \in \mathcal{F}, \theta \in \Theta\}$. In practice, the functions are observed on a grid $T = \{t_1, ..., t_D\} \subset \mathcal{T}$. This extends the manifold learning formalization to

$$\Theta \xrightarrow{\phi} \mathcal{M}_\mathcal{F} \xrightarrow{e} \mathcal{Y}. \tag{4.6}$$

Setting, for example, $\Theta = \mathbb{R}$ and $x(t) = \theta_1 t + \theta_2$, $\theta_1, \theta_2 \in \Theta$, we have a well accessible functional manifold and can sample a (D-dimensional) data set $X = \{x_1(t), ..., x_n(t)\} \subset \mathcal{M}_\mathcal{F}$ of n functional observations with the specified intrinsic structure accordingly. Combined with the good visualization capability, this makes functional data extremely useful to qualitatively evaluate, analyze, and compare manifold learning approaches and results. We make extensive use of these aspects in Chapters 7, 8, and 9.

4.5. Reliability in Unsupervised Learning

4.5.1. A Fundamental Problem

In unsupervised learning "the machine simply receives inputs $x_1, x_2, ...$, but obtains neither supervised target outputs, nor rewards from its environment" (Ghahramani, 2004, p. 74). In particular, it is usually assumed the inputs are i.i.d. observations following a distribution P, but there is no vector of outputs $y_1, ..., y_n$ as in supervised learning. Unsupervised learning encompasses tasks such as clustering, outlier detection, (nonlinear) dimension reduction, and manifold learning to name the examples most relevant for this work.
Just as in supervised learning, reliability of results is a general issue but the problem here is much more involved. In particular, approaches to systematic benchmark studies are still in their infancy (Van Mechelen et al., 2018). To illustrate the extent of the problem, a few lengthy passages from the literature on the subject are collected here and reproduced verbatim. Hastie et al. (2009, p. 487) state:

> "In the context of supervised learning, there is no such direct measure of success. It is difficult to ascertain the validity of inferences drawn from the output of most unsupervised learning algorithms. One must resort to heuristic arguments not only for motivating algorithms as is often the case in supervised learning as well, but also for judgments as to the quality of results. This uncomfortable situation has led to heavy proliferation of proposed methods, since effectiveness is a matter of opinion and cannot be verified directly."

In contrast, Zimmermann (2020, p. 3) emphasizes:

> "Many unsupervised pattern mining algorithms are rarely, if ever, evaluated on additional data after they have been published. Clustering algorithms are often evaluated time and again on the same data sets, typically in comparison to newer techniques. Algorithms are rarely extensively compared against each other."

Finally, Van Mechelen et al. (2018, p. 2) point out that

> "in this domain there is much less of a benchmarking tradition. This is, for instance, evidenced by the fact that very often new methods are proposed without a sound comparison with predecessors, which obviously seriously hampers a cumulative building of knowledge. Also, within the clustering domain there is a dearth of recommendations and guidelines for benchmarking."

So, while Zimmermann (2020) and Van Mechelen et al. (2018) emphasize a general lack of proper method comparisons, Hastie et al. (2009) point out the fundamental underlying problem that there is no "direct measure of success" and that thus performance assessment and method comparison is, in general, much less objective in unsupervised learning. This makes overoptimistic and non-reliable findings even more likely in unsupervised learning. Consider the following examples that illustrate this problem.
In a study comparing the two manifold learning methods UMAP and t-SNE in single-cell data, Becht et al. (2019) conclude that UMAP better preserves the global structure of data sets than t-SNE and underpin this with extensive and elaborate experiments. Yet, Kobak & Linderman (2021) show in a follow-up that these conclusions only follow because t-SNE was randomly initialized, while UMAP used a PCA-based initialization. They conclude that "there is currently no evidence that the UMAP algorithm per se has any advantage over t-SNE

in terms of preserving global structure" (Kobak & Linderman, 2021, p. 156). Note that well reproducible experiments by Becht et al. (2019) enabled Kobak & Linderman (2021) to pin this down, which again demonstrates the importance of reproducibility but also that reproducibility alone is not a sufficient condition for reliability.
In a study entitled "DBSCAN Revisited: Mis-Claim, Un-Fixability, and Approximation", Gan & Tao (2015) claim that the run time complexity of the well-established clustering method DBSCAN does not hold. They propose a new method and demonstrate that it is more efficient. However, in a follow-up Schubert et al. (2017) in turn point out inaccuracies in that study and show that the new method does not lead to practical improvements because DBSCAN's hyperparameters were poorly specified in the experiments by Gan & Tao (2015), again emphasizing how easily misspecified methods can lead to unreliable conclusions.
The last example goes in a slightly different direction. Bojchevski et al. (2018) proposed *NetGan*, a generative model to generate graphs that are satisfactorily similar to real-world examples. Yet, Rendsburg et al. (2020) show that the generative adversarial network (GAN) is essentially unnecessary and that the same goal can be achieved by a much simpler approach leveraging the crucial transition matrix approximation step only. As Rendsburg et al. (2020, p. 1) emphasize, "being much simpler on the conceptual side, we reveal the implicit inductive bias of the algorithm — an important step towards increasing the interpretability, transparency and acceptance of machine learning systems".
So, while the first two examples illustrate that proper performance assessment and method comparison are hard to achieve in unsupervised learning regardless of the complexity of the methods, the third example points to the more subtle aspect that a lack of clear understanding of the underlying problem may lead to more complex methods than necessary. We provide some more background on these aspects in the following.

4.5.2. Methodological and Conceptual Aspects

The fundamental problem in unsupervised learning is the lack of an outcome y that would allow to clearly define a learning objective (such as the ERM criterion reflected in Eq. 3.1) to optimize. In other words, there is no ground truth to compare the results of unsupervised learning methods against (Shalev-Shwartz & Ben-David, 2014; Zimmermann, 2020). This affects the entire analysis chain as described in Chapter 3 from performance assessment, through model selection, to method comparison and benchmark studies.

Performance Assessment How to properly assess performance and evaluate results is the most important question and a lot of work has been devoted to this question. For *outlier detection*, see, for example, Schubert et al. (2012), Goix (2016), Marques et al. (2020); for *clustering* see Rand (1971), Ben-David & Ackerman (2008), Vinh et al. (2010), Rendón et al. (2011), Ullmann et al. (2022); and for *manifold learning* see Lee & Verleysen (2008), Lee & Verleysen (2009), L. Chen & Buja (2009), Rieck & Leitte (2015), Kraemer et al. (2018), Liang et al. (2020). In general, external evaluation and internal evaluation criteria need to be distinguished in cluster analysis. The external criteria make use of external information about the data, in particular, class labels, and compare the partition obtained by a clustering method with the partition induced by the external information. Internal measures, in contrast, compute absolute or average intra- and inter-cluster distances of a obtained partition. The external evaluation approach is also very common in outlier detection where observations from a class are chosen as inliers and contaminated by observations from another class. In general, this approach needs to assume that external information reflects the inherent structure of the

data sufficiently, i.e., it constitutes a meaningful ground truth for the problem at hand, and may lead to misleading conclusions otherwise (e.g., see Campos et al., 2016; Luxburg et al., 2012; Van Mechelen et al., 2018). In manifold learning, the performance is often evaluated by comparing the overlap of neighborhoods in the low-dimensional embedding to neighborhoods in the high-dimensional observation space. What would be of major interest is the reconstruction error, which, however, is not accessible for many manifold learning methods as they do not provide an embedding function explicitly (see Section 4.2).

Hyperparameter Selection and Method Comparison Moreover, some authors have paid particular attention to how to set hyperparameters. For example, Thomas et al. (2016) develop a general tuning approach for unsupervised outlier detection methods based on the area-under-the-mass-volume-curve (AUMVC). However, the computational complexity increases with the dimension of the data set as it requires Monte-Carlo integration. It is thus not well applicable to high-dimensional data. Alaız (2015) proposes to select hyperparameters of the manifold learning method *Diffusion Maps* using neighborhood preservation measures introduced by Lee & Verleysen (2009). In contrast, Belkina et al. (2019) propose an approach for automatic selection of t-SNE's hyperparameters within the context of single cell data.
Finally, notwithstanding the general lack of comparison studies (Van Mechelen et al., 2018; Zimmermann, 2020), there have been efforts in this direction as well, see, for example, Campos et al. (2016), Goldstein & Uchida (2016), and Domingues et al. (2018) for outlier detection, and Y. Wang et al. (2021) and Cantini et al. (2021) for manifold learning. In their *white paper*, Van Mechelen et al. (2018) even provide general guidelines for benchmarking in clustering and discuss several examples.
Despite these efforts, Zimmermann (2020) nevertheless emphasizes that it remains mostly unclear (1) how to evaluate whether a method's results reflect the relevant structures of the underlying data generating process, (2) how to select the various hyperparameters many unsupervised learning methods are adjusted to a specific setting with, (3) how to compare different methods and how to decide on the superiority of one over the other. He concludes that there "is therefore need for more, and more extensive, evaluations in both pattern mining and clustering" (Zimmermann, 2020, p. 3).

Conceptualizations Moreover, note that there are approaches toward general conceptualizations and frameworks. For example, outlier detection has been formalized based on minimum level sets (Scott & Nowak, 2006) and M-estimation (Clémençon & Jakubowicz, 2013). In principle, this means that outliers are defined as objects in low-density regions of the distribution assumed to generate the data.
In contrast, Kleinberg (2002) provides an impossibility theorem that suggests that deriving a unifying framework for clustering is complicated: in particular, he shows that no clustering function fulfills scale-invariance, richness, and consistency, three fundamental properties of a clustering function according to the author. Scale-invariance means that a clustering function is robust to changes in the unit used to measure distances between observations, while richness implies that for a set of observations all possible partitions can be the result of a given clustering function. Finally, consistency requires that increasing inter-cluster distances and decreasing intra-cluster distances does not change the clustering result. However, in a follow-up, Ben-David & Ackerman (2008) shift the focus from clustering functions to quality criteria. Based on another set of axioms that clustering quality criteria should fulfill, they show that a consistent conceptualization of clustering is possible. Further examples in this direction include a conceptualization of linkage-based clustering (Ackerman et al., 2010) and

clustering approaches in general (Carlsson & Mémoli, 2013). In manifold learning, Agrawal et al. (2021) provide a very general framework called *minimum-distortion embedding* that provides a unifying perspective on different embedding methods, but also semi-supervised learning and sphere packing, for example.
On the other hand, despite the described efforts, there is also evidence of considerable conceptual ambiguity in these areas (recall the examples from outlier detection, cluster analysis, and manifold learning provided in Chapter 2). This partially explains why different research communities treat these problems – for example clustering or outlier detection – more or less in isolation: there is (so far) no established common ground the problems can be traced back to. Consider, for example, that in functional data analysis outlier detection has been a highly investigated research topic in recent years and many complex methods that are highly specific to functional data have been developed. Potentially useful approaches from the pattern mining community, however, have not gained much attention as pointed out in Chapter 8. Similarly, clustering is a highly investigated problem in the pattern mining community as well as in topological data analysis and manifold learning, yet there appear to be only loose connections between these areas as pointed out in Chapter 10.
In summary, to improve reliability in unsupervised machine learning, two aspects appear crucial: (1) more extensive and systematic method evaluations and comparisons and (2) improving conceptual clarity and systematic underpinnings. The contributions in Part III are devoted to these issues with a specific focus on manifold learning and we provide a structural overview on the contribution in the following.

4.5.3. Structural Overview of Part III

As outlined, the major part of the thesis is concerned with unsupervised learning from a manifold learning perspective. That means, Part III has a much broader scope than Part II. To make the overall context and relationship to reliability more accessible, it is helpful to structure the contributions in Part III in advance according to their specific methodological and application-oriented aspects on the one side, and a common, more general conceptual aspect on the other side.
First of all, within the individual contributions, we (1) elaborate on methodological aspects of manifold learning as a specific unsupervised learning task itself. For example, we investigate tuning approaches for manifold learning methods based on surrogate performance measures. On the other hand, we (2) also use manifold learning as a kind of auxiliary procedure to provide insights on methodological and conceptual issues in other data analysis tasks (functional data analysis, outlier detection, clustering). These conceptual aspects are specifically related to the individual tasks. For example, we argue that two types of outliers, *structural* and *distributional*, have to be distinguished and we use concepts from manifold learning to make this explicit. Moreover, we argue that settings with clearly separable clusters should be more strictly differentiated from those where clusters (are allowed to) overlap. Again this is demonstrated based on concepts and approaches from manifold learning. Recall that we consider improving *conceptual clarity* a crucial part of improving reliability. We argue that these contributions can reduce some of the conceptual ambiguity present in the specific tasks. However, these specific conceptual aspects considered within the individual contributions are not to be confused with a more general, overarching conceptualization we intend to more clearly establish. This general conceptualization is (implicitly) developed across the individual contributions and allows to more precisely trace the individual, task-specific problems back to a common ground. In the following, we shortly outline this general conceptualization, as it can be seen as a common thread running through the individual papers as a whole. The exact

implications, however, will likely become fully tangible only after considering the individual contributions. Therefore, we provide a concluding summary on this general aspect in Part IV and postpone a more detailed discussion until then.

We argue that, for the problem of outlier detection, different intrinsic data structures will be relevant than for the problem of cluster analysis (or for manifold learning, at least based on the standard assumption), but that these structures are usually not made explicit enough. As a consequence, some of the described conceptual ambiguity surrounding these tasks may be reduced if the relevant structures were made more precise. We show that a more general notion of manifold learning that goes beyond the standard assumption that there is a single, connected manifold, allows for a more precise conceptualization of these tasks. For that, we refer to the underlying structures as *inner* geometry, *outer* geometry, and the *topology* of a data set (cf. Lee & Verleysen, 2007, Ch. 7.4; Tenenbaum et al., 2000). In particular, we argue that the *inner* geometry is specifically relevant for the standard notion of manifold learning, *outer* geometry is specifically relevant for outlier detection, and the *topology* for clustering. The outer geometry requires the notion of an ambient or surrounding space to correctly reflect the relevant data structures. This is particularly relevant for outlier detection because we assume that *structural* or *off-manifold* outliers stem from a different manifold than the bulk of the observations. Therefore, the spatial position and distances in the ambient space must be inferred and retained to reflect the outlier structure. The inner geometry, on the other hand, does not require the notion of an ambient space. Assuming a single, connected manifold, the goal is to infer the structure of this manifold. In both cases, however, isometry (i.e. preserving distances) is important. Yet, for problems where the topology of a data set is particularly important, for example, in cluster detection, neither structure induced by some ambient space nor the specific (intrinsic) structure of a manifold is of major interest. In contrast, leveraging topological features such as connected components is important. The crucial point is that not all aspects may be accessible in a suitable manner at once. In particular, when obtaining low-dimensional representation, some structures, for example, outliers, likely get lost if the representation of other structures, for example, topological aspects, is prioritized.

In summary, the central theme in Part III is *manifold learning*. Yet, the contributions also focus on *functional data* (analysis), *outlier detection*, and *cluster analysis* in particular. Again, the overarching goal is to work towards more reliability in these areas. The contributions in Part III elaborate on these aspects as follows.

Chapter 7 investigates how suitable embedding quality measures as described in Section 4.3.3 are for *hyperparameter tuning* of manifold learning methods. From a conceptual perspective, we concentrate on data stemming from a single connected manifold, i.e., settings where the *inner geometry* is of interest. In doing so, we leverage the favorable properties of *functional data* outlined in Section 4.4 to assess the tuning approach.

Chapters 8 and 9 are concerned with *outlier detection*. Chapter 8 again focuses on the application to *functional data*. Assuming that there are two fundamental and distinct outlier types, we demonstrate that a conceptualization based on two manifolds much better reflects the problem. That means, we are in a setting where the *outer geometry* is of most interest and we demonstrate that this improves functional outlier detection conceptually and practically. Chapter 9 then generalizes the approach to other data types such as images or graphs. Moreover, we review concepts and terminology from the literature indicating a general conceptual ambiguity. We show that the provided conceptualization based on two manifolds can considerably advance conceptual clarity in outlier detection in general.

Finally, Chapter 10 focuses on *cluster analysis*. We show that focusing on (parts of) the *topology* of the data set is crucial for this task. In particular, we demonstrate – applying the clustering method DBSCAN (Ester et al., 1996) to UMAP embeddings – that leveraging the unconnected components of a data set can considerably enhance clustering.

Part II.

Supervised Benchmarking

5. Large-scale Benchmark Study of Survival Prediction Methods Using Multi-omics Data

Chapter 5 describes a benchmark study of 13 survival prediction methods that are applied to 18 cancer data sets from 'The Cancer Genome Atlas' (TCGA). These high-dimensional data sets consist of five feature groups including clinical and four different types of molecular features.

Contributing article:

Herrmann, M., Probst, P., Hornung, R., Jurinovic, V., & Boulesteix, A. L. (2021). Large-scale benchmark study of survival prediction methods using multi-omics data. Briefings in bioinformatics, 22(3), bbaa167. https://doi.org/10.1093/bib/bbaa167

Copyright information:

Author contributions:

Moritz Herrmann implemented the experiment and wrote the manuscript. Philip Probst and Roman Hornung assisted in the implementation. Vindi Jurinovic gathered the raw data. Anne-Laure Boulesteix came up with the initial idea. All authors jointly revised the manuscript.

Supplementary material available at:

Code and data: https://github.com/HerrMo/multi-omics_benchmark_study

Note on originality:

The presented study is an extension of the Master's thesis by Moritz Herrmann. As part of the Master's thesis, the basic study design was developed, the data prepossessed, and a preliminary experiment implemented and conducted. The additional accomplishments undertaken as part of this dissertation are as follows:

First of all, three additional methods – SGL, blockForest, GRridge – were included (and implemented). Note that implementing GRridge led to a bug fix (see https://github.com/markvdwiel/GRridge/issues/2). The (extended) experiment was then completely rerun and re-evaluated. In particular, *model failures* and differences in *computation time* were examined more closely. In addition, statistical tests were carried out and confidence intervals calculated. Furthermore, reproducibility and neutrality were considerably increased: (1) the existing code was thoroughly reworked, restructured, and re-implemented, (2) code and data were made available on GitHub and OpenML, respectively, and (3) for the methods GRridge, SGL, glmboost, CoxBoost, and ranger the developers were contacted and asked for an evaluation of the implementation and setup (e.g., this resulted in an extensive adaptation of the GRridge procedure).

Briefings in Bioinformatics, 22(3), 2021, 1–15

doi: 10.1093/bib/bbaa167
Method Review

Large-scale benchmark study of survival prediction methods using multi-omics data

Moritz Herrmann, Philipp Probst, Roman Hornung, Vindi Jurinovic and Anne-Laure Boulesteix

Corresponding author: Moritz Herrmann, Department of Statistics, Ludwig Maximilian University, Munich, 80539, Germany. Tel: +49 89 2180 3198; E-mail: moritz.herrmann@stat.uni-muenchen.de.

Abstract

Multi-omics data, that is, datasets containing different types of high-dimensional molecular variables, are increasingly often generated for the investigation of various diseases. Nevertheless, questions remain regarding the usefulness of multi-omics data for the prediction of disease outcomes such as survival time. It is also unclear which methods are most appropriate to derive such prediction models. We aim to give some answers to these questions through a large-scale benchmark study using real data. Different prediction methods from machine learning and statistics were applied on 18 multi-omics cancer datasets (35 to 1000 observations, up to 100 000 variables) from the database 'The Cancer Genome Atlas' (TCGA). The considered outcome was the (censored) survival time. Eleven methods based on boosting, penalized regression and random forest were compared, comprising both methods that do and that do not take the group structure of the omics variables into account. The Kaplan–Meier estimate and a Cox model using only clinical variables were used as reference methods. The methods were compared using several repetitions of 5-fold cross-validation. Uno's C-index and the integrated Brier score served as performance metrics. The results indicate that methods taking into account the multi-omics structure have a slightly better prediction performance. Taking this structure into account can protect the predictive information in low-dimensional groups—especially clinical variables—from not being exploited during prediction. Moreover, only the block forest method outperformed the Cox model on average, and only slightly. This indicates, as a by-product of our study, that in the considered TCGA studies the utility of multi-omics data for prediction purposes was limited.
Contact: moritz.herrmann@stat.uni-muenchen.de, +49 89 2180 3198
Supplementary information: Supplementary data are available at *Briefings in Bioinformatics* online. All analyses are reproducible using R code freely available on Github.

Key words: multi-omics data; prediction models; benchmark; survival analysis; machine learning; statistics

Moritz Herrmann is a PhD student at the department of statistics, University of Munich (Germany). His research interests include computational statistics, machine learning and functional data analysis.
Philipp Probst obtained a PhD in statistics from the University of Munich, working at the Institute for Medical Informatics, Biometry and Epidemiology (Germany). His research interests include machine learning, statistical software development and benchmark experiments.
Roman Hornung is a postdoctoral fellow at the Institute for Medical Informatics, Biometry and Epidemiology, University of Munich (Germany). His research interests include biostatistics, computational statistics and machine learning.
Vindi Jurinovic is a postdoctoral fellow at the Institute for Medical Informatics, Biometry and Epidemiology, University of Munich (Germany). Her research interests include biometry, bioinformatics and computational molecular medicine.
Anne-Laure Boulesteix is an associate professor at the Institute for Medical Information Processing, Biometry and Epidemiology, University of Munich (Germany). Her activities mainly focus on computational statistics, biostatistics and research on research methodology.
Submitted: 7 March 2020; **Received (in revised form):** 25 June 2020

Introduction

In the past two decades, high-throughput technologies have made data stemming from molecular processes available on a large scale ('omics data') and for many patients. Starting from the analysis of whole genomes, other molecular entities such as mRNA or peptides have also come into focus with the advancing technologies. Thus, various types of omics variables are currently under investigation across several disciplines such as genomics, epigenomics, transcriptomics, proteomics, metabolomics and microbiomics [1].

It may be beneficial to include these different data types in models predicting outcomes, such as the survival time of patients. Until recently, only data from a single omics type were used to build such prediction models, with or without the inclusion of standard clinical data [2]. In recent years, however, the increasing availability of different types of omics data measured for the same patients (called multi-omics data from now on) has led to their combined use for building outcome prediction models. An important characteristic of multi-omics data is the high-dimensionality of the datasets, which frequently have more than 10 000 or even 100 000 variables. This places particular demands on the methods used to build prediction models: they must be able to handle data where the number of variables by far exceeds the number of observations. Moreover, practitioners often prefer sparse and interpretable models containing only a few variables [3]. Last but not the least, multi-omics data are structured: the variables are partitioned into (nonoverlapping) groups. This structure may be taken into account when building prediction models.

Several methods have been specifically proposed to handle multi-omics data, while established methods for high-dimensional data from the fields of statistics and machine learning also seem reasonable for use in this context. Although there are studies with a limited scope comparing some of these methods, there has not yet been a large-scale systematic comparison of their pros and cons in the context of multi-omics using a sufficiently large amount of real data.

The pioneering study by Bøvelstad *et al.* [4] investigates the combined use of clinical and one type of molecular data, using only four datasets. In one of the first studies devoted to methodological aspects of multi-omics-based prediction models, Zhao *et al.* [5] compare a limited number of methods for multi-omics data based on a limited number of datasets. Lang *et al.* [6] investigate automatic model selection in high-dimensional survival settings, using similar but fewer prediction methods than our study. Moreover, again only four datasets are used. A study by De Bin *et al.* [7] investigates the combination of clinical and molecular data, with a focus on the influence of correlation structures of the feature groups, but it is based on simulated data.

Our study aims to fill this gap by providing a large-scale benchmark experiment for prediction methods using multi-omics data. It is based on 18 cancer datasets from The Cancer Genome Atlas (TCGA) and focuses on survival time prediction. We use several variants of three widely used modeling approaches from the fields of statistics and machine learning: penalized regression, statistical boosting and random forest. The aim is to assess the performances of the methods and the different ways to take the multi-omics structure into account. As a by-product of our study, we also obtain results on the added predictive value of multi-omics data over models using only clinical variables.

The remainder of the paper is structured as follows. The Methods section briefly outlines the methods under investigation. In the subsequent Benchmark experiment section, we describe the conducted experiment. The findings are presented in the Results section, which is followed by a discussion.

Methods

Preliminary remarks

There are essentially two ways to include multi-omics data in a prediction model. The first approach, which we term as naive, does not distinguish the different data types, i.e. does not take the group structure into account. In the second approach, the group structure is taken into account. The advantage of the naive approach, its simplicity, comes at a price. First of all, physicians and researchers often have some kind of prior knowledge of which data type might be especially useful in the given context [3]. If so, it is desirable to include such information by incorporating the group structure. Well-established prognostic clinical variables which are known to be beneficial for building prediction models for a specific disease are an important special case. In this situation, it may be useful to take the group structure into account during model building or even to treat clinical variables with priority. Otherwise, these clinical variables might get lost within the huge amount of omics data [2]. To some extent, the same might be true for different kinds of omics data. If, for example, gene expression (rna) is expected to be more important than copy number variation (cnv) data for the purpose of prediction, it might be useful to incorporate the distinction between these two data types into the prediction model or even to prioritize rna in some sense.

Other important aspects of prediction models from the perspective of clinicians are sparsity, interpretability and transportability [3]. Methods yielding models which are sparse with regard to the number of variables and number of omics types are often considered preferable from a practical perspective. Interpretation and practical application of the model to the prediction of independent data are easier with regression-based methods yielding coefficients that reflect the effects of variables on the outcome than with machine learning algorithms [8].

Finally, in addition to the prediction performances of the different methods, the question of the additional predictive value of omics data compared with clinical data is also interesting from a clinical perspective [2]. Many of the omics-based prediction models which were claimed to be of value for predicting disease outcomes could eventually not be shown to outperform clinical models in independent studies [2, 4, 9]. However, some findings suggest that using both clinical and omics variables jointly may outperform clinical models [4, 10, 11]. In our benchmark study, we can address this issue by systematically comparing the performance of clinical models and combined models for a large number of datasets.

The methods included in our study can be subsumed in three general approaches, which are briefly described in the following subsections: penalized regression-based methods, boosting-based methods and random forest-based methods. A more technical description of the methods can be found in the supplementary material. It should be noted, however, that several multi-omics specific penalized regression-based methods have already been developed and have readily available implementations in R, while the same is not true for the other two classes of methods to the same extent. Consequently, there is an imbalance in the

number of methods included for each class. Moreover, our study does not include deep learning approaches. To the best of our knowledge, studies using deep learning based on multi-omics data mostly focus on classification. For the two approaches we are aware of that successfully applied deep learning on multi-omics data to predict survival times [12, 13]—the latter not yet formally published— there are at the moment no thorough, established and easy to use implementations in R. Similar to extended boosting methods, we did not include deep learning approaches for these reasons.

Moreover, two reference methods are considered: simple Cox regression, which only uses the clinical variables, and the Kaplan–Meier estimate, which does not use any information from the predictor variables.

Penalized regression-based methods

The penalized regression methods briefly reviewed in this section have in common that they modify maximum partial likelihood estimation by applying a regularization, most importantly to account for the $n << p$ problem.

Standard Lasso, introduced more than two decades ago [14] and subsequently extended to survival time data [15], applies L_1-regularization to penalize large (absolute) coefficient values. The result is a sparse final model: a number of coefficients are set to zero. The number of nonzero coefficients decreases with increasing penalty parameter λ and cannot exceed the sample size. The method does not take the group structure into account. The parameter λ is a hyper-parameter to be tuned.

Two-step (TS) IPF-Lasso [16] is an extension of the standard Lasso specifically designed to take a multi-omics group structure into account. This method is an adaptation of the integrative Lasso with penalty factors (IPF) [17], which consists in allowing different penalty values for each data type. In TS-IPF-Lasso, the ratios between these penalty values are determined in a first step (roughly speaking, by applying standard Lasso and averaging the resulting coefficients).

Priority-Lasso [3] is another Lasso-based method designed for the incorporation of different groups of variables. Often, clinical researchers prioritize variables that are easier, cheaper to measure or known to be good predictors of the outcome. The principle of priority-Lasso is to define a priority order for the groups of variables. Priority-Lasso then successively fits Lasso-regression models to these groups, whereby at each step, the resulting linear predictor is used as an offset for the Lasso model fit to the next group. For the study at hand, however, we do not have any substantial domain knowledge, so we cannot specify a meaningful priority order. We therefore alter the method into a TS procedure similar to the TS IPF-Lasso. More precisely, we order the groups of variables according to the mean absolute values of their coefficients fitted in the first step by separately modeling each group. This ordering is used as a surrogate for a knowledge-based priority order.

Sparse group Lasso (SGL) [18] is another extension of the Lasso, capable of including group information. The method incorporates a convex combination of the standard Lasso penalty and the group-Lasso penalty [19]. This simultaneously leads to sparsity on feature as well as on group level.

Adaptive group-regularized ridge regression (GRridge) [20] is designed to use group specific co-data, e.g. p-values known from previous studies. Multi-omics group structure may also be regarded as co-data, although the method was originally not intended for this purpose. It is based on ridge regression, which uses a L_2-based penalty term. Feature selection is achieved post hoc by exploiting the heavy-tailed distribution of the estimated coefficients, which clearly separates coefficients close to zero from those which are further away [20].

Boosting-based methods

Boosting is a general technique introduced in the context of classification in the machine learning community, which has then been revisited in a statistical context [21]. Statistical boosting can be seen as a form of iterative function estimation by fitting a series of weak models, so-called base learners. In general, one is interested in a function that minimizes the expected loss when used to model the data. This target function is updated iteratively, with the number of boosting steps m_{stop}, i.e. the number of iterations, being the main tuning parameter. This parameter, together with the so-called learning rate, which steers the contribution of each update, also leads to a feature selection property. In this study, we use two different boosting approaches.

Model-based boosting [22], the first variant, uses simple linear models as base learners and updates only the loss minimizing base learner per iteration. The learning rate is usually fixed to a small value such as 0.1 [23].

Likelihood-based boosting [24], in contrast, uses a penalized version of the likelihood as loss and the shrinkage is directly applied in the coefficient estimation step via a penalty parameter. It is also an iterative procedure: the updates of previous iterations are included as an offset to make use of the information gained.

Random forest-based methods

Random forest is a tree-based ensemble method introduced by Breiman [25]. Instead of growing a single classification or regression tree, it uses bootstrap aggregation to grow several trees and aggregates the results. Random forest was later expanded to survival time data [26]. For each split in each tree, the variable maximizing the difference in survival is chosen as the best feature. Eventually, the cumulative hazard function is computed via the Nelson–Aalen estimator in each final node in each tree. For prediction, these estimates are averaged across the trees to obtain the ensemble cumulative hazard function.

Block forest [27] is a variant modifying the split point selection of random forest to incorporate the group structure (or 'block' structure, hence the name of the method) of multi-omics data. It can be applied to any outcome type for which a random forest variant exists.

Benchmark experiment

Study design

Our study is intended as a neutral comparison study; see [28, 29] for an exact definition and discussions of this concept. Firstly, we compare methods that have been described elsewhere and do not aim at emphasizing a particular method. Secondly, we tried to achieve a reasonable level of neutrality, which we disclose here following the example of Couronné *et al.* [30]. As a team, we are approximately equally familiar with all classes of methods. Some of us have been involved in the development of priority-Lasso, IPF-Lasso and block forest. As far as the other methods are concerned, we contacted the person listed in CRAN as package maintainer via email and asked for an evaluation of our implementation including the choice of parameters.

A further important aspect of the study design is the choice and number of datasets used for the comparison, since the performance of prediction methods usually strongly varies across datasets. Boulesteix *et al.* [29] compare benchmark experiments to clinical trials, where methods play the role of treatments and datasets play the role of patients. In analogy to clinical trials, the number of considered datasets should be chosen large enough to draw reliable conclusions, and the selection of datasets should follow strict inclusion criteria and not be modified after seeing the results; see the Datasets section for more details on this process. Finally, a benchmark experiment should be easily extendable (and, of course, reproducible). It is almost impossible to include every available method in a single benchmark experiment, and it should also be easy to compare methods proposed later without re-running the full experiment and without too much programming effort. For this reason, we use the R package *mlr* [31], which offers a unified framework for benchmark experiments and makes them easily extendable and reproducible.

Technicalities and implementation

The benchmark experiment is conducted using R 3.5.1 [32]. We compare the 13 learners described in the Method configurations section on 18 datasets (see the Datasets section). The code to reproduce the results is available on GitHub (https://github.com/HerrMo/multi-omics_benchmark_study), the data can be obtained from OpenML [33, 34] (https://www.openml.org/). To further improve reproducibility, the package *checkpoint* [35] is used. Because the computations are time demanding but parallelisable, the package *batchtools* [36] is used for parallelisation. The package *mlr* [31], used for this benchmark experiment, offers a simple framework to conduct all necessary steps in a unified way.

The focus of our study is the general performance of prediction methods. In this context, cross-validation (CV) is a standardly used procedure to obtain estimates of the prediction performance of a prediction method when applied to data with similar characteristics as the training data. We use 10×5-fold CV for datasets with a size less than 92 MB (11 datasets) and 5×5-fold CV for datasets with a size larger than 112 MB (7 datasets) to keep computation times feasible.

The proportion of patients with events is very small for some datasets. We avoid training sets with few or even zero events using stratification by event status, i.e. the resulting training sets have comparable censoring rates. Moreover, hyperparameter tuning is performed. This could in principle also be implemented via *mlr*, but in this study, the tuning procedures provided by the specific packages are used. We denote the resampling strategy used for hyper-parameter tuning inner-resampling and the repeated CV used for performance assessment outer-resampling. For inner-resampling we use out-of-bag (OOB) samples for random forest learners and 10-fold CV for the other learners.

Performance evaluation

The performance is evaluated in three dimensions. First of all, the prediction performance is assessed via the integrated Brier score and the C-index suggested by Uno *et al.* [37] (hereinafter simply denoted as ibrier and cindex). The time range for calculation is set to the maximum event time of the individual CV test set. While the cindex only measures discriminative power, the ibrier also measures calibration. Moreover, the cindex, unlike the ibrier, is not a strictly proper scoring rule [38]. The ibrier should therefore be used as the primary measure for prediction accuracy. If, however, one is interested in ranking patients according to their risk, the cindex is also a valid measure. Ranking the patients according to their risks is relevant from a practical point of view because in many applications of risk prediction, the goal is to assign the patients to fixed, ordered risk classes. Another reason we included the cindex as a secondary measure in our study is that it is routinely used as a standard measure in benchmarking, thus allowing easier comparison with other studies.

The second dimension is the sparsity of the resulting models, which has two aspects: sparsity on the level of variables and sparsity on group level. The latter refers to whether variables of only some groups are selected. Sparsity on feature level, in contrast, refers to the overall sparsity, i.e. the total number of selected features. As random forest does not perform variable selection, it is not assessed in this dimension. Computation times are considered as a third dimension.

Another important aspect is the different use of group structure information. Some of the methods do not use any such information, some favor clinical data over molecular data, and some differentiate between all groups of variables (i.e. also between omics groups). Thus, the differences in performance might not only result from using different prediction methods. They may also arise from the way in which the group structure information is included. Therefore, comparability in terms of predictive performance is only given for methods that use the same strategy to include group information: (i) naive methods not using the group structure; (ii) methods using the group structure and not favoring clinical features; (iii) methods using the group structure by favoring clinical features, where we subsume methods favoring clinical and not distinguishing molecular covariates *and* methods favoring clinical and additionally also distinguishing molecular covariates.

Method configurations

Following the terminology of the package *mlr* [31], we denote a method configuration as a 'learner'. There may be several learners based on the same method. An overview of learners considered in our benchmark study is displayed in Table 1, while the full specification is given in the paragraph devoted to the corresponding method. In the following, the R packages used to implement the learners can be found in parentheses after the paragraph heading.

Penalized regression-based learners

Lasso (glmnet [39, 40]). The penalty parameter λ is chosen via internal 10-fold CV. No group structure information is used.

SGL (SGL [41]). The model is fit via the *cvSGL* function. Tuning of the penalty parameter λ is conducted via internal 10-fold CV. The parameter α steering the contribution of the group-Lasso and the standard Lasso is not tuned and set to the default value 0.95, as recommended by the authors [18]. All other parameters are set to default as well.

TS IPF-Lasso (ipflasso [42]). The penalty factors are selected in the first step by computing separate ridge regression models for every feature group and averaging the coefficients within the groups by the arithmetic mean. These settings have shown reasonable results [16]. The choice of the penalty parameters λ_m

Table 1. Summary of learners used for the benchmark experiment.

Learner	Method	Package::function	Tuning
Lasso	Standard Lasso	glmnet::cv.glmnet	10-f-CV
ipflasso*	TS IPF-Lasso	ipflasso::cvr.ipflasso	10-f-CV
prioritylasso*	Priority-Lasso	priortiylasso::prioritylasso	10-f-CV
prioritylasso favoring*	Priority-Lasso	priortiylasso::prioritylasso	10-f-CV
grridge*	GRridge	GRridge::grridge	10-f-CV
SGL*	SGL	SGL::cvSGL	10-f-CV
glmboost	Model-based boosting	mboost::glmboost	10-f-CV
CoxBoost	Likelihood-based boosting	CoxBoost::cv.CoxBoost	10-f-CV
CoxBoost favoring*	Likelihood-based boosting	CoxBoost::cv.CoxBoost	10-f-CV
ranger	Random forest	tuneRanger::tuneMtryfast	OOB
blockForest*	Block forest	blockForest::blockfor	OOB
Clinical only	Cox model	survival::coxph	No
Kaplan–Meier	Kaplan–Meier estimate	survival::survival	No

The use of group structure information is indicated with *.

is conducted using 5-fold-CV in the first step and 10-fold CV in the second.

Priority-Lasso (prioritylasso [43]). The priority order is determined through a preliminary step realized in the same way as in the first step of TS IPF-Lasso. The priority-Lasso method takes into account the group structure. Even though the version with cross-validated offsets delivers slightly better prediction results [3], the offsets are not estimated via CV in order to not increase the computation times further. To select the parameter λ in each step of priority-Lasso, 10-fold CV is used.

Priority-Lasso favoring clinical features (prioritylasso [43]). The settings are the same as before, except that the group of clinical variables is always assigned the highest priority. The preliminary step only determines the priority order for the molecular groups. The clinical variables are used as an offset when fitting the model of the second group. Furthermore, the clinical variables are not penalized (setting parameter *block1.penalization = FALSE*).

GRridge (GRridge [44]). This method was not originally intended for the purpose of including multi-omics group structure but is capable of doing so. To better fit the task at hand, a special routine was provided by the package author in personal communication. In addition, the argument *selectionEN* is set to *TRUE* so post-hoc variable selection is conducted, and *maxsel*, the maximum number of variables to be selected, is set to 1000.

Boosting-based learners

Model-based boosting (mboost [45]). Internally, *mlr* uses the function *glmboost* from the package *mboost* and sets the family argument to *CoxPH()*. Furthermore, the number of boosting steps (m_{stop}) is chosen by a 10-fold CV on a grid from 1 to 1000 via *cvrisk*. For the learning rate ν the default value of 0.1 is used. Group structure information is not taken into account.

Likelihood-based boosting (CoxBoost [46]). The maximum number of boosting steps *maxstepno* is set to default, i.e. 100. Again, m_{stop} is determined by 10-fold CV. The penalty λ is set to default and thus computed according to the number of events. No group structure information is used.

Likelihood-based boosting favoring clinical features (CoxBoost [46]). The settings are the same as before. Additionally, group structure information is used by specifying the clinical features as mandatory. These features are favored as in the case of priority-Lasso by setting them as an offset and not penalizing them. Further group information is not used, so the molecular data are not distinguished.

Random forest-based learners

Random forest (ranger [48]). Tuning of *mtry* is conducted via the *tuneMtryfast* function of package *tuneRanger*. The minimal node size is 3 (the ranger default settings). The other hyperparameters are set to default as well. Note that we also investigated the *randomForestSRC* [47] implementation of random forest, which leads to comparable results, but worked only on Windows and not on Ubuntu if parallelization was conducted via package *batchtools*. We thus show only the results obtained with *ranger*.

Block forest (blockForest [49]). Block forest is a random forest variant able to include group structure information. The implementation is based on *ranger*. With function *blockfor* the models are fit via the default settings.

Reference methods

The clinical reference model is a Cox proportional hazard model, computed via the *coxph* function of the *survival* package [50] and only uses clinical features. The Kaplan–Meier estimate is computed via *survfit* from the same package.

Datasets

From the cancer datasets that have been gathered by the TCGA research network (http://cancergenome.nih.gov), we selected those with more than 100 samples and five different multi-omics groups, which resulted in a collection of datasets for 26 cancer types (a list of these 26 cancer types is provided in the supplement). As described below, further preprocessing eventually lead to 18 usable datasets. Table 2 gives an overview of these 18 datasets and the abbreviations used to reference them within the study.

For each cancer type, there are four molecular data types and the clinical data type, i.e. five groups of variables. The molecular data types comprise cnv, rna, miRNA expression (mirna) and mutation. It should be noted that the choice of data types was motivated by practical issues to a certain extent. In particular, methylation data would have been interesting to consider but could not be included due to their massive size, which would have led to long downloading and computation times. However,

Table 2. Summary of the datasets used for the benchmark experiment. The third to the seventh column show the number of features in the feature group, the eighth column the total amount of features (p). The last three columns show, in this order, the number of observations (n), the number of effective cases (n_e) and the ratio of the number of events and the number of observations (r_e).

Dataset	Cancer	Clin.	cnv	mirna	mutation	rna	p	n	n_e	r_e
BLCA	Bladder urothelial	5	57964	825	18577	23081	100455	382	103	0.27
BRCA	Breast invasive C.	8	57964	835	17975	22694	99479	735	72	0.10
COAD	Colon adenocarcinoma	7	57964	802	18538	22210	99524	191	17	0.09
ESCA	Esophageal C.	6	57964	763	12628	25494	96858	106	37	0.35
HNSC	Head–neck squamous CC.	11	57964	793	17248	21520	97539	443	152	0.34
KIRC	Kidney renal clear CC.	9	57964	725	10392	22972	92065	249	62	0.25
KIRP	Cervical kidney RP. CC.	6	57964	593	8312	32525	99403	167	20	0.12
LAML	Acute myeloid leukemia	7	57962	882	2176	29132	90162	35	14	0.40
LGG	Low grade glioma	10	57964	645	9235	22297	90154	419	77	0.18
LIHC	Liver hepatocellular C.	11	57964	776	11821	20994	91569	159	35	0.22
LUAD	Lung adenocarcinoma	9	57964	799	18388	23681	100844	426	101	0.24
LUSC	Lung squamous CC.	9	57964	895	18500	23524	100895	418	132	0.32
OV	Ovarian cancer	6	57447	975	13298	24508	96237	219	109	0.50
PAAD	Pancreatic AC.	10	57964	612	12392	22348	93329	124	52	0.42
SARC	Sarcoma	11	57964	778	10001	22842	91599	126	38	0.30
SKCM	Skin cutaneous M.	9	57964	1002	18593	22248	99819	249	87	0.35
STAD	Stomach AC.	7	57964	787	18581	26027	103369	295	62	0.21
UCEC	Uterine corpus EC.	11	57447	866	21053	23978	103358	405	38	0.09

Abbreviations: C. indicates carcinoma; CC., cell carcinoma; PP, renal papilla; AC., adenocarcinoma; M., melanoma; EC., endometrial carcinoma.

such compromises for practical reasons are inevitable and other data types could have been considered.

The number of variables differs strongly between groups but is similar across datasets. Most molecular features (about 60 000) belong to the cnv group but only a few hundred features to mirna, the smallest group. There is a total of about 80 000 to 103 000 molecular features for each cancer type.

Of the 26 available datasets, three were excluded because they did not have observations for every data type. Furthermore, since the outcome of interest is survival time, not only the number of observations is crucial but, most importantly, the number of events (deaths), which we call the number of effective cases. A ratio of 0.2 of effective cases is common [10]. The five datasets that had less than 5% effective cases were excluded.

Since the majority of the clinical variables had missing values, the question arose of which to include for a specific dataset while saving as many observations as possible. As we did not have any domain knowledge, we adopted a two-step strategy. Firstly, an informal literature search was conducted to find studies where the specific cancer type was under investigation. Variables mentioned to be useful in these studies were included, if available. Secondly, we additionally used variables that were available for most of the cancer types. These comprised sex, age, histological type and tumor stage. These were included as standard, if available. Of course, sex was not included for the sex-specific cancer types.

Finally, note that our study considers only one dataset per cancer type, i.e. per prediction problem. It evaluates the candidate learners using CV, which reasonably estimates the prediction performance that would be obtained for a population with similar characteristics (in statistical terms, with similarly distributed data). From a practical clinical perspective, it is crucial to evaluate prediction rules on independent external data before implementing them in practice. The resulting estimated prediction performance is usually less impressive than the CV estimate [51]. However, although some discrepancies between the rankings can be observed, the difference between rankings is less prominent than the difference between errors. We assume that methods performing best in CV also range among the best ones when applied to external datasets, which is compatible with the extensive results presented in Bernau *et al.* [52].

Results

Failures and refinement of the study design

As a consequence of the repeated 5-fold CV, $11 \cdot 10 \cdot 5 + 7 \cdot 5 \cdot 5 = 725$ models are fit for each learner in total. Some of these model fittings—i.e. some CV iterations—were not successful. This is common in benchmark experiments of larger scale [53]. Such a failure does not affect the other 4 CV iterations and the remaining nine repetitions of CV but leads to missing values for the assessment measures for the failing iterations, which have to be handled when computing averaged measures. To cope with such modeling failures, we follow strategies described previously [53, 54]. If a learner fails in more than 20% of the CV iterations for a given dataset, we assign (for the failing iterations) values of the performance measures corresponding to random prediction (0.25 for ibrier and 0.5 for cindex) and the mean of the other iterations for the computation time and the number of selected features. If a learner fails in less than 20%, the performance means of the successful iterations are assigned for all measures. See Table 4 for the learners' failure rates averaged over the datasets.

Modeling failures are mainly learner related, that is, there are no datasets for which many or all learners are unstable, but individual learners are particularly unstable. For every dataset, there are at least seven learners yielding no failures. In contrast, learners Lasso, CoxBoost and glmboost are rather unstable, with 31.2%, 28.6% and 19.7% failures in total. IPF-Lasso is more stable overall, but also considerably unstable for specific datasets. The other learners are very stable, with not a single failure for the random forest variants, the reference methods and CoxBoost favoring. Dataset specific failure rates can be taken from the tables in the supplement, which show the learner performances like in Table 4 broken down by dataset.

Table 3. Performance of SGL on four small datasets. Column 'All' represents the total number of selected features, the subsequent columns the numbers of selected features of the respective groups.

Data	cindex	ibrier	Time	All	clin	cnv	mirna	rna	mut
LAML	0.496	0.231	1.9	8149	0.5	7822	4.7	0	323
LIHC	0.533	0.198	9.0	3617	0.3	3250	28	264	75
PAAD	0.650	0.255	4.5	1483	3.2	62	30	12	1375
SARC	0.629	0.278	7.5	3081	2.7	1906	51	40	1082
mean	0.58	0.24	5.7	4083	1.7	3260	28	79	714

Besides such modeling failures, more general issues related to usability occurred while conducting the experiment. First of all, using SGL with the considered large numbers of features always leads to a fatal error in R under Windows, but not using the Linux distribution Ubuntu 14.04. More importantly, the extremely long computation times for SGL were problematic. Since we received no feedback from the authors, we used the standard settings. These lead to computations lasting several days for one single model fit for large datasets. Altering some of the parameters did not strongly reduce the computational burden. Running the whole experiment as planned was thus not possible for SGL. Here, we briefly present the results of SGL which could be obtained based on four of the smallest datasets. For the rest of the study we exclude SGL from the analysis. On average, over all iterations and the four datasets, SGL leads to a cindex of 0.58 and an ibrier of 0.24. The resulting models are neither sparse on feature level, with an average of 4083 selected features, nor on group level. The mean computation time of 5.7 h for one CV iteration confirms the problem of extremely long computation times. In comparison, the next slowest method needs 1.2 h for one iteration, on average over all datasets. Table 3 shows the performance values of SGL for each of the four datasets and the mean values.

Computation time

Table 4 shows the average performance measures for every method and is ordered by the ibrier. All values are obtained by—firstly—averaging over the outer-resampling CV iterations and—secondly—averaging over the datasets. For the methods not yielding model coefficients the corresponding cells contains '-'.

The ninth column of Table 4 displays the mean computation time. The computation times are measured as the time needed for model fitting (training time). The fastest procedures are standard Lasso and ranger, followed by glmboost and the CoxBoost variants. The three penalized regression methods using group structure (IPF-Lasso, priority-Lasso, GRridge) are about two to three times slower, with GRridge being the fastest of the three methods. Of the two prioritylasso variants the one favoring clinical features is a little slower. Finally, blockForest is the slowest method.

Of course, the computation times depend on the size of the datasets. Figure 1 displays the mean computation time of one outer-resampling iteration for the different learners and datasets. The datasets are ordered from smallest (LAML) to largest (BRCA).

The long computation times of priority-Lasso and IPF-Lasso for COAD and KIRP are notable. COAD and KIRP are among the smaller datasets with 17 (9%) and 20 (12%) events, respectively. Generally, computation times vary more over the CV iterations for these methods. For KIRP and COAD this variation is particularly strong, with individual model fittings taking up to 50 h. However, the CV iterations which lead to these extreme computation times for IPF-Lasso and priority-Lasso, lead to modeling failures for the unstable learners Lasso, glmboost and CoxBoost. That means, especially priority-Lasso and to lesser extent IPF-Lasso are more stable for particular CV iterations than the unstable learners, but this comes at the price of increased computation times. Note, moreover, all Lasso variants rely on the same implementation of Lasso (glmnet). The more specific methods improve the Lasso approach in terms of stability, because where standard Lasso fails, often the more specific variants do not.

Model sparsity

To assess sparsity, the number of nonzero coefficients of the resulting model of each CV iteration is considered. As random forest models do not yield such coefficients, this aspect is not assessed for random forest variants.

Sparsity on the level of variables

Sparsity in terms of the number of included variables is particularly interesting for practical purposes, since sparse models are easier to interpret and to communicate. On average, as Table 4 shows, ipflasso leads to the sparsest models with an average of seven variables, followed by CoxBoost with on average 10 variables. CoxBoost favoring and Lasso are also reasonably sparse (13, 16), but the variability is higher for Lasso (Figure 2). glmboost, prioritylasso and prioritylasso favoring yield models with more than 20 features (22, 26, 30). Least sparse is grridge; the average grridge model size (984) is close to the maximum number of features to be selected (*maxsel* = *1000*). grridge seems not to be able to appropriately select variables in this setting (recall that it is not intended to do so).

Sparsity on group level

Table 4 (see also Figure 1 in the supplement) shows that grridge and priority-Lasso choose variables from all groups and are thus not sparse on group level. Among the other methods, IPF-Lasso yields strong sparsity on group level. Mostly clinical features are selected. Furthermore, with boosting variants CoxBoost and glmboost and with standard Lasso no clinical features are selected. CoxBoost favoring does not select mirna features. IPF-Lasso does not include cnv and rna features. This exemplifies the problem of methods treating high- and low-dimensional groups equally. As already pointed out, due to their low dimension, clinical variables get lost within the huge number of molecular variables. It becomes obvious that this also applies for some of the molecular variables. The mirna group is, in comparison with the other molecular groups, lower dimensional with 585 to 1002 features. Learners which do not consider group structure fail to include clinical variables and include at most one mirna

Table 4. Average learner performances. The values are obtained by averaging over the CV iterations and datasets. The time is measured in minutes. The second column shows the affiliation of the methods using the multi-omics data to the categories described in section 3.3. Column 'All' represents the total number of selected features, the subsequent columns the numbers of selected features of the respective groups. For learners not yielding model coefficients, the corresponding measures are set to "-". The 'ci' columns display the 95% confidence intervals for the means based on quantiles of the t-distribution; observations are the learners' average values over the CV iterations, i.e. one observation per dataset. The total number of features may differ from the sum of features in each group due to rounding errors. The last column depicts the overall percentage of modeling failures.

		ibrier			cindex										
Learner	Category	Mean	sd	ci	Mean	sd	ci	Time	All	clin	cnv	mima	rna	mut	failures in %
blockForest	Non-fav	0.174	0.042	[0.153, 0.195]	0.620	0.072	[0.584, 0.656]	80	–	–	–	–	–	–	0
CoxBoost favoring	Fav	0.174	0.036	[0.156, 0.192]	0.618	0.057	[0.589, 0.646]	13	13	8	1	0	3	1	0
Clinical only	–	0.175	0.038	[0.156, 0.194]	0.618	0.060	[0.588, 0.648]	0	8	8	–	–	–	–	0
CoxBoost	Naive	0.175	0.039	[0.155, 0.194]	0.552	0.080	[0.512, 0.592]	14	10	0	1	1	5	2	28.6
ipflasso	Non-fav	0.176	0.034	[0.159, 0.193]	0.578	0.100	[0.528, 0.628]	33	7	5	0	1	0	1	9.7
ranger	Naive	0.179	0.045	[0.157, 0.202]	0.562	0.068	[0.529, 0.596]	10	–	–	–	–	–	–	0
prioritylasso	Non-fav	0.180	0.037	[0.162, 0.199]	0.591	0.068	[0.558, 0.625]	32	26	4	4	7	7	4	1.2
Kaplan–Meier	–	0.180	0.040	[0.160, 0.200]	0.500	0.000	[0.500, 0.500]	0	–	–	–	–	–	–	0
prioritylasso favoring	Fav	0.181	0.040	[0.161, 0.201]	0.607	0.056	[0.579, 0.635]	39	30	8	4	6	6	5	1.9
grridge	Non-fav	0.181	0.044	[0.159, 0.203]	0.587	0.069	[0.553, 0.622]	28	984	2	332	19	540	91	1.4
glmboost	Naive	0.188	0.037	[0.169, 0.206]	0.542	0.104	[0.490, 0.593]	12	22	0	3	1	10	7	19.7
Lasso	Naive	0.198	0.034	[0.181, 0.215]	0.546	0.089	[0.502, 0.591]	8	16	0	2	1	8	5	31.2

variable. CoxBoost favoring, which differentiates clinical and molecular variables, does not select mirna features. In contrast, learners taking into account the multi-omics group structure generally include variables of both lower dimensional groups. Using the group structure thus prevents low-dimensional groups from being discounted.

Prediction performance

Overview and main findings

Figure 2 shows the distributions of the values of the performance metrics across the datasets. Again, grridge is excluded from the sparsity panel. Three important findings can be highlighted. First of all, regarding Figure 2, most of the learners perform better than the Kaplan–Meier estimate (indicated by the dashed horizontal line). This indicates that using the variables is, in general, useful. Only Lasso performs worse than the Kaplan–Meier estimate (based on the ibrier). Secondly, only blockForest outperforms the reference clinical Cox model (red horizontal line), which stresses the importance of the clinical variables for these datasets. Finally, methods taking into account the group structure in some way in general outperform the naive methods.

Comparing prediction methods

All analyses in this section refer to Table 4.

Naive methods The learners CoxBoost, glmboost, ranger and standard Lasso are fit with the naive strategy. In general, the results are not consistent over the two measures. With regards to the ibrier, likelihood-based boosting performs best. Moreover, model-based boosting performs better than Lasso but gets outperformed by random forest which is close to likelihood-based boosting. According to the cindex, random forest performs best followed by likelihood-based boosting and Lasso. Model-based boosting performs the worst. All methods are at least slightly better than the Kaplan–Meier estimate. To sum up, although the results differ depending on the considered measure, random forest shows a tendency to outperform the other methods, since it is among the best methods based on the ibrier and performs best based on the cindex.

Methods not favoring clinical features The learners block forest, ipflasso, prioritylasso and grridge use the group structure but do not favor clinical features. The random forest variant blockForest outperforms the other methods. It performs better on average than any other method based on both measures. Among the penalized regression methods, IPF-Lasso performs best according to the ibrier and priority-Lasso according to the cindex. GRridge ranks third according to the ibrier and second according to the cindex. Moreover, priority-Lasso and GRridge perform equal to or even worse than the Kaplan–Meier estimate based on the ibrier. Since IPF-Lasso yields the sparsest models, it might be preferable when sparsity is important.

Methods favoring clinical features There are two learners favoring clinical features: CoxBoost favoring and prioritylasso favoring. The results are unambiguous with CoxBoost favoring performing better than prioritylasso favoring. Furthermore, both learners perform better than the Kaplan–Meier estimate based on the cindex, but only CoxBoost favoring performs better than the Kaplan–Meier estimate based on the ibrier. Thus, according to these findings, likelihood-based boosting yields better results than priority-Lasso when clinical variables are favored, even though priority-Lasso here further distinguishes the molecular data.

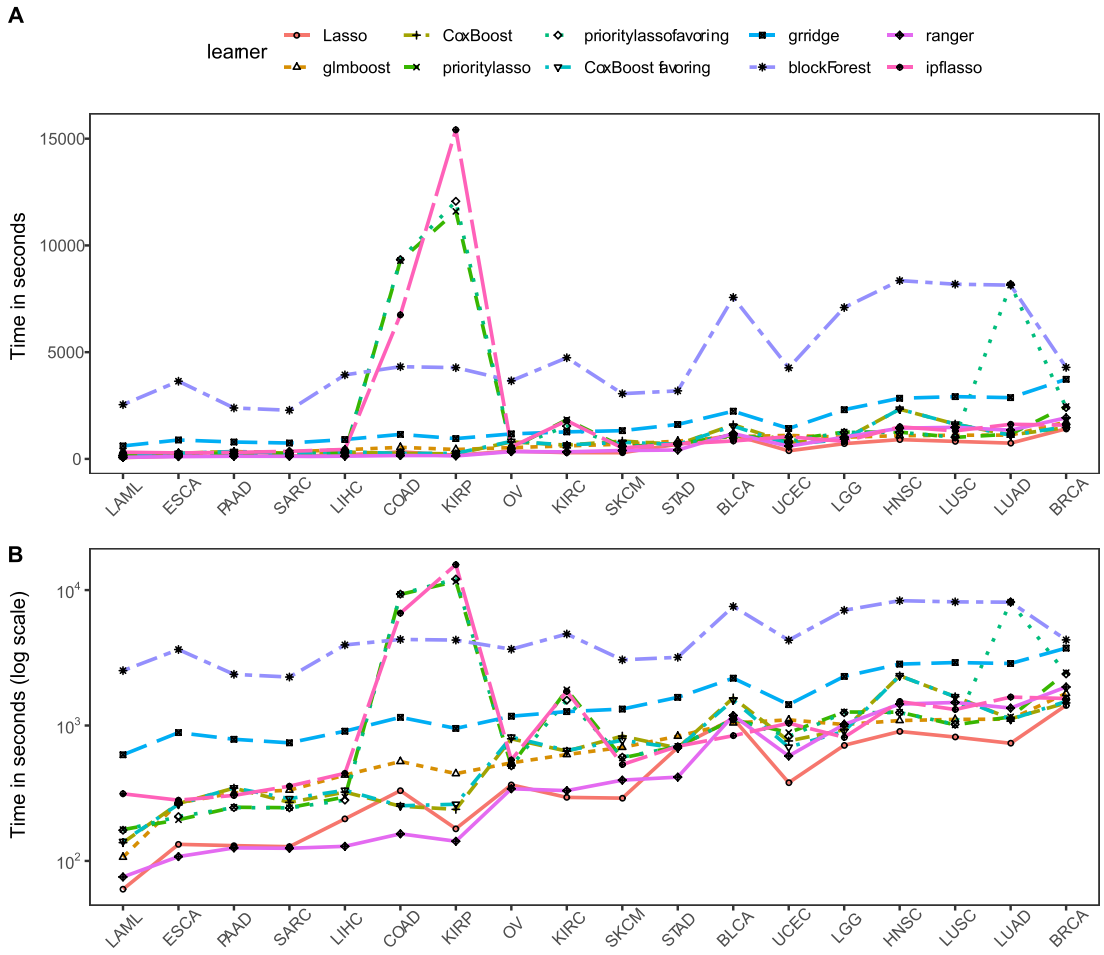

Figure 1. Computation time. (A) Computation times in seconds. (B) Computation times in log(seconds). The datasets are ordered form smallest (LAML) to largest (BRCA).

However, when comparing the described performances—especially if based on the aggregated results in Table 4—it is very important to consider the heterogeneity over the datasets. As the standard deviations and confidence intervals in Table 4 and the boxplots in Figure 2 show, the performance of the methods varies strongly over the datasets (the confidence intervals are obtained by using the learners' average values over the CV iterations as observations, i.e. one observation per dataset). The dataset specific learner performances are depicted in the supplementary tables. Moreover, paired two-sided *t*-tests comparing, for example, blockForest with CoxBoost favoring and the clinical only model show no statistically significant differences in performance with *p*-values of 0.81 and 0.86 (cindex) and 0.95 and 0.78 (ibrier). For all *t*-tests in the study, the normal distribution assumption was checked with Shapiro–Wilk tests and Q-Q plots. Thus, considering the small differences in performance and the variability of the method performances across datasets, conclusions about the superiority of one method over another should be treated with great caution.

Using multi-omics data

In the first part of this section, we summarize our results regarding the added predictive value of multi-omics as a by-product of our benchmark study, before eventually focusing on the methodological comparison of the different approaches of taking the structure of the clinical and multi-omics variables into account.

Added predictive value To assess the added predictive value of the molecular data, we follow approach A proposed by Boulesteix and Sauerbrei [2], thus comparing learners obtained by only using clinical features and combined learners, i.e. learners using clinical and molecular variables. Since it is emphasized that for this validation approach the combined learners should not be derived by the naive strategy, these learners are not considered here.

In general, the findings indicate that the multi-omics data may have the potential to add predictive value. First of all, blockForest outperforms the Cox model based on both measures. Secondly, as Table 5 shows, there are several datasets for which there is at least one learner that takes the group structure into account and outperforms the clinical learner. For some of the

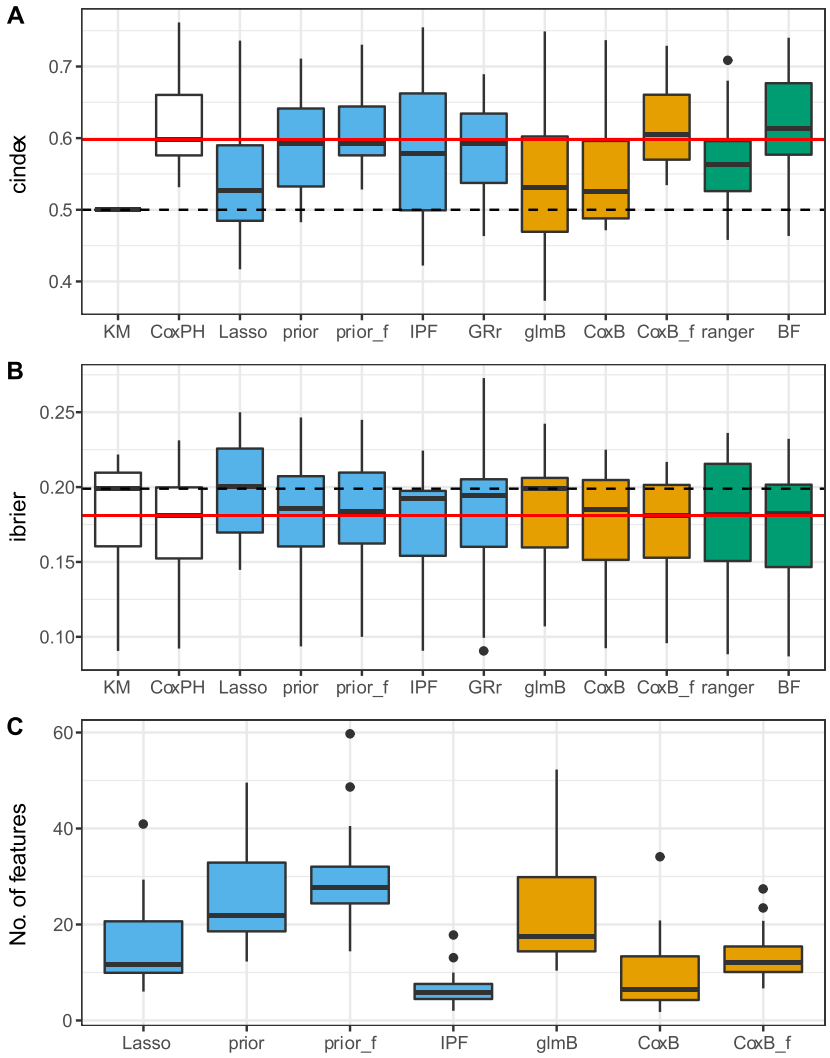

Figure 2. Performance of the learners. A: cindex. B: ibrier. C: total number of selected features; only learners yielding model coefficients are included and grridge is excluded since it yields models on a much larger scale. The solid red and dashed black horizontal lines correspond to the median performance of the clinical only model and the Kaplan–Meier estimate, respectively. Colors indicate membership to one of the general modeling approaches: penalized regression (blue), boosting (orange), random forest (green), reference methods (white). Abbreviations: KM indicates Kaplan–Meier; Lasso, Lasso; glmB, glmboost; CoxB, CoxBoost; CoxPH, clinical only; prior, priority-Lasso; prior_f, priority-Lasso favoring; IPF, ipflasso; CoxB_f, CoxB favoring; GRr, grridge; BF, blockForest; ranger, ranger.

datasets, e.g. LAML and COAD, the performance differences are substantial. Thus, using additional molecular data leads to better prediction performances in some of the considered cases. On the other hand, it must be taken into account that in the other cases the differences are small and, again regarding the confidence intervals, one has to be careful when drawing conclusions about the superiority over the Cox model. Moreover, for six datasets the Cox model does not get outperformed by methods which use the omics data. This raises serious concerns regarding a beneficial effect of the omics data as far as the considered TCGA data are concerned.

Including group structure In general, the results suggest that using the naive strategy of treating clinical and molecular variables equally leads to a worse performance in comparison to methods that take the group structure into account. Table 6 shows the mean performance of the naive learners and the structured learners (both favoring and not favoring the clinical features) by dataset. That is, we compare methods using multi-omics data and additionally its structure (structured learners: ipflasso, priortylasso variants, grridge, CoxBoost favoring, blockForest) against methods using the multi-omics data but not taking into account its structure (naive learners: Lasso, ranger, CoxBoost, glmboost). The clinical only model and the Kaplan–Meier estimate are not considered here. Each value is computed as average over the naive respectively the structured learners' mean cindex and ibrier values. Only in five cases is

Table 5. Assessment of the added predictive value of the molecular variables by dataset. The second and seventh column show the best performing learners for the respective dataset and measure, the 'cindex' and 'ibrier' columns the performances of these learners. In the cases where the clinical only model is outperformed, the 'Ref.' columns show the corresponding cindex and ibrier values of the reference Cox model only using clinical variables. The "ci" columns show the 95% confidence intervals for the respective performance values based on quantiles of the t-distribution; observations are the learners' CV iteration values. Note that these intervals are intended to give a notion of the stability of the mean values, but are—in contrast to Table 4—not valid confidence intervals, since the CV iterations are no independent observations [55, 56]. Bold letters indicate datasets for which there is a method using the group structure and outperforming the Cox model for both measures.

Data	Learner	ibrier	ci	Ref.	ci	Learner	cindex	ci	Ref.	ci
BLCA	CoxBoost favoring	0.190	[0.181, 0.199]	0.192	[0.183, 0.201]	CoxBoost favoring	0.640	[0.612, 0.668]	0.633	[0.607, 0.659]
BRCA	blockForest	0.141	[0.134, 0.149]	0.147	[0.137, 0.158]	CoxBoost favoring	0.643	[0.618, 0.669]	0.637	[0.608, 0.666]
COAD	blockForest	0.087	[0.075, 0.099]	0.101	[0.088, 0.115]	blockForest	0.656	[0.586, 0.725]	0.541	[0.475, 0.608]
ESCA	ipflasso	0.209	[0.198, 0.221]	0.214	[0.199, 0.228]	Clinical only	0.574	[0.536, 0.612]	–	–
HNSC	glmboost	0.202	[0.193, 0.211]	0.210	[0.201, 0.220]	blockForest	0.582	[0.554, 0.610]	0.554	[0.519, 0.588]
KIRC	ipflasso	0.144	[0.138, 0.149]	0.146	[0.140, 0.152]	Clinical only	0.761	[0.734, 0.789]	–	–
KIRP	ranger	0.118	[0.106, 0.131]	0.140	[0.117, 0.163]	grridge	0.629	[0.566, 0.692]	0.572	[0.502, 0.641]
LAML	ranger	0.182	[0.165, 0.199]	0.231	[0.200, 0.263]	ranger	0.709	[0.651, 0.766]	0.596	[0.534, 0.657]
LGG	Lasso*	0.145	[0.132, 0.157]	0.168	[0.154, 0.181]	glmboost	0.749	[0.719, 0.779]	0.652	[0.618, 0.685]
LIHC	ranger	0.146	[0.135, 0.157]	0.169	[0.158, 0.180]	grridge	0.602	[0.560, 0.645]	0.586	[0.542, 0.630]
LUAD	CoxBoost favoring*	0.172	[0.160, 0.183]	0.172	[0.161, 0.183]	prioritylasso	0.665	[0.640, 0.690]	0.663	[0.631, 0.695]
LUSC	grridge	0.210	[0.203, 0.217]	0.216	[0.205, 0.227]	prioritylasso favoring	0.537	[0.502, 0.572]	0.531	[0.502, 0.561]
OV	ipflasso*	0.169	[0.163, 0.174]	0.173	[0.167, 0.179]	prioritylasso	0.600	[0.582, 0.618]	0.598	[0.580, 0.617]
PAAD	Clinical only*	0.190	[0.178, 0.202]	–	–	prioritylasso favoring	0.686	[0.658, 0.714]	0.683	[0.655, 0.712]
SARC	glmboost*	0.179	[0.167, 0.190]	0.202	[0.188, 0.217]	blockForest	0.685	[0.651, 0.720]	0.673	[0.637, 0.709]
SKCM	Clinical only	0.191	[0.185, 0.198]	0.191	[0.185, 0.198]	blockForest	0.597	[0.556, 0.639]	0.581	[0.540, 0.623]
STAD	Clinical only	0.192	[0.182, 0.202]	–	–	Clinical only	0.598	[0.555, 0.641]	–	–
UCEC	ipflasso	0.091	[0.079, 0.102]	0.092	[0.080, 0.105]	Clinical only	0.686	[0.581, 0.791]	–	–

*indicates that there is a method with equal performance.

Table 6. Comparing naive learners and structured learners. The performance of structured learners, i.e. learners using the group structure, and naive learners are compared for every dataset. The cindex and ibrier columns show the mean performance values for the corresponding dataset and learner types. Bold values indicate better values for the given dataset.

	ibrier		cindex	
Data	Structured	Naive	Structured	Naive
BLCA	**0.198**	0.201	**0.618**	0.595
BRCA	**0.152**	0.187	**0.598**	0.512
COAD	**0.104**	0.120	**0.518**	0.480
ESCA	0.235	**0.234**	**0.506**	0.477
HNSC	0.210	0.210	**0.562**	0.557
KIRC	**0.154**	0.156	**0.721**	0.690
KIRP	**0.132**	0.136	**0.560**	0.532
LAML	**0.207**	0.217	**0.634**	0.558
LGG	0.169	**0.153**	0.695	**0.726**
LIHC	0.171	**0.167**	**0.566**	0.559
LUAD	**0.181**	0.194	**0.636**	0.539
LUSC	**0.220**	0.229	**0.501**	0.457
OV	**0.172**	0.192	**0.575**	0.448
PAAD	**0.196**	0.203	**0.663**	0.588
SARC	0.197	**0.180**	**0.667**	0.624
SKCM	**0.200**	0.221	**0.580**	0.509
STAD	**0.199**	0.210	**0.556**	0.525
UCEC	**0.103**	0.119	**0.646**	0.538

the average performance of the naive learners better than the average performance of the structured learners: regarding the ibrier, the naive learners perform better than the structured learners for four datasets; regarding the cindex, only for the LGG dataset is the performance of the naive learners higher than the performance of the structured learners. Unpaired, one-sided *t*-tests for the four naive and the six structured learners, using the mean performance values of the individual methods over the datasets as observations, yield *p*-values of 0.1273 and 0.0002 for the ibrier and cindex, respectively.

In summary, if multi-omics data are used—although there is a general concern regarding the usefulness of models using multi-omics compared with simple clinical models—its structure should also be taken into account.

Favoring clinical features According to our findings, favoring clinical variables leads to better prediction results. For likelihood-based boosting, this is in line with the findings of others (see [57] and the reference therein). Differentiating the clinical variables from the molecular features strongly increases the prediction performance of likelihood-based boosting (CoxBoost and CoxBoost favoring), according to the average cindex. Favoring clinical features raises likelihood-based boosting from one of the worst to one of the best performing methods. Moreover, our findings show this might also hold for methods which use the multi-omics group structure. For priority-Lasso the increase is not as strong, but still notable when considering the cindex. Yet, the ibrier does not confirm this.

Discussion

In general, one should be very careful when interpreting the results of our benchmark experiment and drawing conclusions. Most importantly, the findings highly depend on the considered prediction performance measure, as the method ranking changes drastically between the two measures. For example, CoxBoost performs poorly based on the cindex but performs third best regarding the ibrier. These findings indicate that the performance of a method may change dramatically if a different performance measure is used for its assessment. Moreover, according to ibrier, two methods perform better than the Cox model (though only slightly), and six methods perform worse than the Kaplan–Meier estimate. Generally, since the cindex only measures discrimination and is not a strictly proper scoring rule, the ibrier should be considered more important. In particular, if prognostic accuracy is of interest, preference should be given to the ibrier. However, given its interpretability, the cindex could be preferred if risk classification is the main objective.

Another important aspect of the performance assessment is, as shown in Figure 2 and Table 5, the variability across datasets. The superiority of one method over the other strongly depends not only on the considered performance measure but also, most importantly, on the considered datasets. This stresses the importance of large-scale benchmarks, like this one, which use many datasets. Since the variability between datasets is huge, we need many datasets—a fact well known by statisticians performing sample size calculations, which however tends to be ignored when designing benchmark experiments using real datasets [58]. If we had conducted our study with, say, 3, 5 or 10 datasets (as usual in the literature), we would have obtained different—more unstable—results.

Regarding prediction performance, blockForest outperforms the other methods on average overall datasets. Moreover, it is the only method which outperforms the simple Cox model on average regarding both measures. The other methods using the molecular data do not perform better than the simple Cox model. The better prediction performance of blockForest, however, comes at the price of long computation times. Apart from SGL, blockForest is the slowest method. The fastest learners, standard Lasso and ranger, are about 10 times faster and blockForest is still 2 times slower than the next slowest learner prioritylasso favoring. Moreover, like the standard random forest variant, it does not yield easily interpretable models, even though the strengths of the variables can be assessed via the variable importance measure(s) output as a by-product of the random forest algorithms. Thus, taking the other assessment dimensions into account, e.g. CoxBoost favoring clinical features is very competitive. It is quite fast, leads to reasonably sparse models at group and feature level and yields performances only slightly worse than (cindex) or equal to (ibrier) blockForest.

From a practical perspective, even simpler modeling approaches, such as a simple Cox model using clinical variables only, might be preferable. This model is easily interpretable, needs only a fraction of the computation time and, with a mean cindex of 0.618 and a mean ibrier of 0.175, performs only slightly worse than blockForest (0.620 and 0.172) and comparable to or even better than all other methods. Note, however, that blockForest also offers the possibility of favoring clinical covariates using the argument *always.select.block* of the *blockfor* function. Hornung and Wright [27] show that this can improve the prediction performance of block forest considerably. However, since this option was not yet available at the time of conduction of the analyses performed for the current paper, we were not able to consider this block forest variant here.

In general, conclusions about the superiority of one method over the other with respect to the prediction performance must be drawn with caution, as the differences in performance can be very small and the confidence intervals often show a remarkable overlap. Exemplary *t*-tests comparing blockForest with CoxBoost

favoring and the Clinical only model showed no significant differences in performance. Furthermore, we do not believe that the recommendation of a single method is generally appropriate because even if some methods have a better average performance than others, the ranking between methods depends in large part on the specific dataset used. On the other hand, if there is no independent, external dataset available for performance estimation it is also not advisable to try out too many methods in practice because this increases the risk that the maximum of the cross-validated performance estimates is optimistic [59] and correction methods to adjust for this over-optimism are still in their infancy [60]. In this situation, it is likely best to strike a balance by taking into account the learners which perform reasonably well. Our study identifies methods worth taking into closer consideration in practice. Apart from the clinical model, these are all methods which take the multi-omics structure into account or favor the clinical covariates, as on average those methods performed better than the naive methods not using the group structure (note again, however, that we did not observe statistically significant differences). A method should then be chosen based on the methodology described in the paper using the dataset at hand.

More generally, it should be noted that the choice of a method should result from the simultaneous consideration of various aspects beyond performance. If (i) performance is the main criterion, (ii) the model is intended solely as a prediction tool, and implemented, say, as a shiny application [61], and (iii) sparsity and interpretability are not considered important, blockForest is certainly a very good choice. Other methods may prove attractive in different situations. Finally, let us note that one of the methods that did not perform very well in the present study in terms of performance, priority-Lasso, may perform better in practice when accurate prior knowledge on the groups of variables is available, and allows the user to favor some of the groups—a dimension that could not be taken into account in our comparison study.

A potential limitation of our study is that the datasets were already used by Hornung and Wright [27] in their comparison study. Since they selected the most promising blockForest based on this comparison study, our results may be slightly optimistic regarding the performance of blockForest—a bias mechanism that has been previously described [62]. More precisely, Hornung and Wright [27] initially considered five different variants of random forest taking the block structure into account and identified the best-performing variant using a collection of 20 datasets including the 18 considered in our study. They named this best-performing variant 'block forest'. It is in theory possible that part of the superiority of the selected 'block forest' variant on the specific 20 datasets is due to chance. In this case, our study, which uses 18 of these datasets again, would (slightly) favor block forest. However, this over-optimism only exists if a different one of the five Block Forest variants compared in the Block Forest paper were the best in reality, i.e. if the superiority of the Block Forest variant considered in our paper were just the result of random fluctuations in the paper of Hornung and Wright [27]. Given the fact that Hornung and Wright [27] used a lot of datasets this is rather unlikely and it is plausible that block forest is indeed the best option, in which case the over-optimistic effect for our study can be ruled out. In addition, although our study is based on data from the same cancer studies, there are several notable differences. Hornung and Wright [27] included two additional datasets and did not use the same sets of clinical variables as in our study. Furthermore, to reduce the computational burden, they used only a subset of 2500 variables when groups had more than 2500 variables. Taking all these aspects into account, it is unlikely that our study is noticeably biased, although such a bias is possible in principle. Regarding the advantage of favoring the clinical variables, it is important to note that it strongly depends on the level of predictive information contained in these variables. If clinical variables contain less information than for the datasets used in our analysis, favoring of these covariates might be less useful than they were found to be in this study, or even detrimental. While we strongly recommend considering favoring the clinical variables, this should not necessarily be performed by default.

Another limitation of our study is that it is based on CV—as usual in the context of large-scale benchmarking. We assume that methods performing best in CV also range among the best ones when applied to external datasets, which is compatible with the results presented in Bernau *et al.* [52]. However, in principle, differences may occur. For example, the performance of methods tending to strongly overfit the training data is expected to drop more when considering external data instead of using CV than the performance of methods that do not strongly overfit. These subtle issues could be addressed in future studies using the recently suggested cross-study validation technique. This approach, however, requires the availability of several external datasets for each cancer type that include exactly the same prediction and outcome variables and consider comparable target populations. Currently, such data is simply not available.

Extending the benchmark to further methods (e.g. methods that do not rely on the proportional hazards assumption, which are only represented by random forest in our study) and further data pre-processing approaches as well as further datasets are desirable. In the same vein, it may be interesting to consider alternative procedures to handle model failures in the outer-resampling process, which may lead to different results. There is to date no widely used standardized approach to deal with missing values in this context. This issue certainly deserves further dedicated research. This benchmark experiment is designed such that such extensions are easy to implement. Using the provided code, further methods can be compared to the ones included in this study.

Key Points

- For the collection of the datasets used in this study, the standard Cox model only using clinical variables is very competitive compared with complex methods using multi-omics data. Among the investigated complex methods, only block forest outperforms the Cox model on average over all datasets, and the difference is not statistically significant.
- If multi-omics data is used, its structure should be used as well: on average it increases the predictive performance and prevents low-dimensional feature groups from being discounted.
- Favoring clinical variables over molecular data increases the prediction performance of the investigated methods on average.
- In general, the findings indicate that assessing and comparing prediction methods should be based on a large number of datasets to reach robust conclusions.
- Aside from the main results of the study, we also observed that using multi-omics data can improve

the performance of prediction methods for particular datasets, but the average performance was not improved for the data investigated in our study.

Supplementary Data

Supplementary data are available online at https://academic.oup.com/bib.

Acknowledgements

First of all, we thank the reviewers for their careful evaluation of our study, their valuable comments and their helpful suggestions, which contributed to a significant improvement of the quality of the paper. Moreover, we thank Mark van de Wiel, Benjamin Hofner, Marvin Wright and Harald Binder for their evaluations of our code and their helpful advice regarding the specific method configurations. We also thank Alethea Charlton for proofreading the manuscript.

Funding

This work was supported by the German Federal Ministry of Education and Research (01IS18036A) and by the German Research Foundation (BO3139/4-3 to A.-L.B., HO6422/1-2 to R.H.). The authors of this work take full responsibilities for its content.

References

1. Hasin Y, Seldin M, Lusis A. Multi-omics approaches to disease. *Genome Biol* 2017;**18**:83–98.
2. Boulesteix A-L, Sauerbrei W. Added predictive value of high-throughput molecular data to clinical data and its validation. *Brief Bioinform* 2011;**12**:215–29.
3. Klau S, Jurinovic V, Hornung R, *et al.* Priority-lasso: a simple hierarchical approach to the prediction of clinical outcome using multi-omics data. *BMC Bioinformatics* 2018;**19**:322.
4. Bøvelstad HM, Nygård S, Borgan Ø. Survival prediction from clinico-genomic models—a comparative study. *BMC Bioinformatics* 2009;**10**:413.
5. Zhao Q, Shi X, Xie Y, *et al.* Combining multidimensional genomic measurements for predicting cancer prognosis: observations from TCGA. *Brief Bioinform* 2014;**16**:291–303.
6. Lang M, Kotthaus H, Marwedel P, *et al.* Automatic model selection for high-dimensional survival analysis. *J Stat Comput Simul* 2015;**85**:62–76.
7. De Bin R, Boulesteix A-L, Benner A, *et al.* Combining clinical and molecular data in regression prediction models: insights from a simulation study. *Brief Bioinform* 2019;bbz136. doi: https://doi.org/10.1093/bib/bbz136.
8. Boulesteix A-L, Janitza S, Hornung R, *et al.* Making complex prediction rules applicable for readers: current practice in random forest literature and recommendations. *Biom J* 2019;**61**:1314–28.
9. De Bin R, Herold T, Boulesteix A-L. Added predictive value of omics data: specific issues related to validation illustrated by two case studies. *BMC Med Res Methodol* 2014;**14**:117.
10. De Bin R, Sauerbrei W, Boulesteix A-L. Investigating the prediction ability of survival models based on both clinical and omics data: two case studies. *Stat Med* 2014;**33**:5310–29.
11. Binder H, Schumacher M. Allowing for mandatory covariates in boosting estimation of sparse high-dimensional survival models. *BMC Bioinformatics* 2008;**9**:14.
12. Chaudhary K, Poirion OB, Lu L, *et al.* Deep learning-based multi-omics integration robustly predicts survival in liver cancer. *Clin Cancer Res* 2018;**24**:1248–59.
13. Chai H, Zhou X, Cui Z, *et al.* Integrating multi-omics data with deep learning for predicting cancer prognosis. *bioRxiv* 2019. https://www.biorxiv.org/content/10.1101/807214v1 (15 June 2020, date last accessed).
14. Tibshirani R. Regression shrinkage and selection via the lasso. *J R Stat Soc Ser B Stat Methodol* 1996;**58**:267–88.
15. Tibshirani R. The lasso method for variable selection in the Cox model. *Stat Med* 1997;**16**:385–95.
16. Schulze G. Clinical outcome prediction based on multi-omics data: extension of IPF-LASSO. 2017. https://epub.ub.uni-muenchen.de/59092/ (17 October 2019, date last accessed).
17. Boulesteix A-L, De Bin R, Jiang X, *et al.* IPF-LASSO: integrative L1-penalized regression with penalty factors for prediction based on multi-omics data. *Comput Math Methods Med* 2017. doi: 10.1155/2017/7691937.
18. Simon N, Friedman JH, Hastie T. A sparse-group lasso. *J Comput Graph Stat* 2013;**22**:231–45.
19. Yuan M, Lin Y. Model selection and estimation in regression with grouped variables. *J R Stat Soc Ser B Stat Methodol* 2006;**68**:49–67.
20. van de Wiel MA, Lien TG, Verlaat W, *et al.* Better prediction by use of co-data: adaptive group-regularized ridge regression. *Stat Med* 2016;**35**:368–81.
21. Friedman JH. Greedy function approximation: a gradient boosting machine. *Ann Statist* 2001;**29**:1189–232.
22. Hothorn T, Bühlmann P. Model-based boosting in high dimensions. *Bioinformatics* 2006;**22**:2828–9.
23. Bühlmann P, Hothorn T. Boosting algorithms: regularization, prediction and model fitting. *Statist Sci* 2007;**22**:477–505.
24. Tutz G, Binder H. Generalized additive modeling with implicit variable selection by likelihood-based boosting. *Biometrics* 2006;**62**:961–71.
25. Breiman L. Random forests. *Mach Learn* 2001;**45**:5–32.
26. Ishwaran H, Kogalur UB, Blackstone EH, *et al.* Random survival forests. *Ann Appl Stat* 2008;**2**:841–60.
27. Hornung R, Wright MN. Block forests: random forests for blocks of clinical and omics covariate data. *BMC Bioinformatics* 2019;**20**:358.
28. Boulesteix A-L, Lauer S, Eugster MJA. A plea for neutral comparison studies in computational sciences. *PLoS One* 2013;**8**:e61562.
29. Boulesteix A-L, Wilson R, Hapfelmeier A. Towards evidence-based computational statistics: lessons from clinical research on the role and design of real-data benchmark studies. *BMC Med Res Methodol* 2017;**17**:138.
30. Couronné R, Probst P, Boulesteix A-L. Random forest versus logistic regression: a large-scale benchmark experiment. *BMC Bioinformatics* 2018;**19**:270.
31. Bischl B, Lang M, Kotthoff L, *et al.* mlr: machine learning in R. *J Mach Learn Res* 2016;**17**:1–5.
32. R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing, 2018. https://www.R-project.org/ (17 October 2019, date last accessed).

33. Vanschoren J, van Rijn JN, Bischl B, *et al.* OpenML: networked science in machine learning. *SIGKDD Explor* 2013;**15**: 49–60.
34. Casalicchio G, Bossek J, Lang M, *et al.* OpenML: an R package to connect to the machine learning platform OpenML. *Comput Statist* 2017;**32**:1–15.
35. Microsoft Corporation. *Checkpoint: Install Packages from Snapshots on the Checkpoint Server for Reproducibility*, 2018. https://CRAN.R-project.org/package=checkpoint (17 October 2019, date last accessed).
36. Lang M, Bischl B, Surmann D. Batchtools: tools for R to work on batch systems. *J Open Source Softw* 2017;**2**:135.
37. Uno H, Cai T, Pencina MJ, *et al.* On the C-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Stat Med* 2011;**30**: 1105–17.
38. Blanche P, Kattan MW, Gerds TA. The c-index is not proper for the evaluation of-year predicted risks. *Biostatistics* 2019;**20**:347–57.
39. Friedman JH, Hastie T, Tibshirani R. Regularization paths for generalized linear models via coordinate descent. *J Stat Softw* 2010;**33**:1–22.
40. Simon N, Friedman JH, Hastie T, *et al.* Regularization paths for Cox's proportional hazards model via coordinate descent. *J Stat Softw* 2011;**39**:1–13.
41. Simon N, Friedman J, Hastie T, *et al. SGL: Fit a GLM (or Cox Model) with a Combination of Lasso and Group Lasso Regularization*, 2018. https://CRAN.R-project.org/package=SGL (17 October 2019, date last accessed).
42. Boulesteix A-L, Fuchs M, Schulze G. *ipflasso: Integrative Lasso with Penalty Factors*, 2019, R package version 1.1. https://CRAN.R-project.org/package=ipflasso (17 October 2019, date last accessed).
43. Klau S, Hornung R. *prioritylasso: Analyzing Multiple Omics Data with an Offset Approach*, 2017. https://CRAN.R-project.org/package=prioritylasso (17 October 2019, date last accessed).
44. van de Wiel MA, Novianti PW. *GRridge: Better Prediction by Use of Co-Data: Adaptive Group-Regularized Ridge Regression*, 2018. https://bioconductor.org/packages/release/bioc/html/GRridge.html (17 October 2019, date last accessed).
45. Hothorn T, Bühlmann P, Kneib T, *et al. mboost: Model-Based Boosting*, 2018. https://CRAN.R-project.org/package=mboost (17 October 2019, date last accessed).
46. Binder H. *CoxBoost: Cox Models by Likelihood Based Boosting for a Single Survival Endpoint or Competing Risks*, 2013. https://CRAN.R-project.org/package=CoxBoost (17 October 2019, date last accessed).
47. Ishwaran H, Kogalur UB. randomForestSRC: Random forests for survival, regression, and classification (rf-src). 2007. https://CRAN.R-project.org/package=randomForestSRC (17 October 2019, date last accessed).
48. Wright MN, Ziegler A. ranger: a fast implementation of random forests for high dimensional data in C++ and R. *J Stat Softw* 2017;**77**:1–17.
49. Hornung R, Wright MN. blockForest: block forests: random forests for blocks of clinical and omics covariate data. 2019. https://CRAN.R-project.org/package=blockForest (17 October 2019, date last accessed).
50. Therneau TM. *survival: A Package for Survival Analysis in S*, 2015. https://CRAN.R-project.org/package=survival (17 October 2019, date last accessed).
51. Castaldi PJ, Dahabreh IJ, Ioannidis JPA. An empirical assessment of validation practices for molecular classifiers. *Brief Bioinform* 2011;**12**:189–202.
52. Bernau C, Riester M, Boulesteix A-L, *et al.* Cross-study validation for the assessment of prediction algorithms. *Bioinformatics* 2014;**30**:i105–12.
53. Bischl B, Schiffner J, Weihs C. Benchmarking local classification methods. *Comput Statist* 2013;**28**:2599–619.
54. Probst P, Wright M, Boulesteix A-L. Hyperparameters and tuning strategies for random forest. *Data Min Knowl Discov* 2019;**9**:e1301.
55. Bengio Y, Grandvalet Y. No unbiased estimator of the variance of K-fold cross-validation. *J Mach Learn Res* 2004;**5**: 1089–105.
56. Fuchs M, Hornung R, Boulesteix A-L, *et al.* On the asymptotic behaviour of the variance estimator of a U-statistic. *J Stat Plan Infer* 2020;**209**:101–11.
57. De Bin R. Boosting in Cox regression: a comparison between the likelihood-based and the model-based approaches with focus on the R-packages CoxBoost and mboost. *Comput Statist* 2016;**31**:513–31.
58. Boulesteix A-L, Hable R, Lauer S, *et al.* A statistical framework for hypothesis testing in real data comparison studies. *Amer Statist* 2015;**69**:201–12.
59. Boulesteix A-L, Strobl C. Optimal classifier selection and negative bias in error rate estimation: an empirical study on high-dimensional prediction. *BMC Med Res Methodol* 2009;**9**:85.
60. Bernau C, Augustin T, Boulesteix A-L. Correcting the optimal resampling-based error rate by estimating the error rate of wrapper algorithms. *Biometrics* 2013;**69**:693–702.
61. Chang W, Cheng J, Allaire JJ, *et al. Shiny: Web Application Framework for R*, 2018. https://CRAN.R-project.org/package=shiny (12 January 2020, date last accessed).
62. Jelizarow M, Guillemot V, Tenenhaus A, *et al.* Over-optimism in bioinformatics: an illustration. *Bioinformatics* 2010;**26**:1990–8.

6. Over-optimism in Benchmark Studies and the Multiplicity of Design and Analysis Options when Interpreting their Results

Chapter 6 presents a study discussing the effects of analysis and design options on the results of benchmark studies. For illustration, it builds on the benchmark study presented in Chapter 5 and extends the analysis of the results therein. An approach deploying multidimensional unfolding is used to assess the impact of the different options.

Contributing article:

Nießl, C., Herrmann, M., Wiedemann, C., Casalicchio, G., & Boulesteix, A. L. (2022). Over-optimism in benchmark studies and the multiplicity of design and analysis options when interpreting their results. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 12(2), e1441. https://doi.org/10.1002/widm.1441

Copyright information:

Author contributions:

Christina Nießl developed the methodology, conducted the formal analysis, and wrote the manuscript. She will use these contributions in her dissertation. Moritz Herrmann contributed to the idea/design of the project and with his expertise and insights on the original results. Chiara Wiedemann contributed by conducting a preliminary analysis and Giuseppe Casalicchio with his expertise in benchmark studies. Anne-Laure Boulesteix made contributions by continuously revising the manuscript and adding ideas. All authors jointly proofread the manuscript.

Supplementary material available at:

Code and data: https://github.com/NiesslC/overoptimism_benchmark

Received: 3 June 2021 | Revised: 30 September 2021 | Accepted: 6 November 2021

DOI: 10.1002/widm.1441

FOCUS ARTICLE

WIREs DATA MINING AND KNOWLEDGE DISCOVERY WILEY

Over-optimism in benchmark studies and the multiplicity of design and analysis options when interpreting their results

Christina Nießl[1] | Moritz Herrmann[2] | Chiara Wiedemann[1] | Giuseppe Casalicchio[2] | Anne-Laure Boulesteix[1]

[1]Institute for Medical Information Processing, Biometry and Epidemiology, Ludwig Maximilians University Munich, Munich, Germany

[2]Department of Statistics, Ludwig Maximilians University Munich, Munich, Germany

Correspondence
Christina Nießl, Institute for Medical Information Processing, Biometry and Epidemiology, Ludwig Maximilians University Munich, Marchioninistr. 15, D-81377 Munich, Germany.
Email: cniessl@ibe.med.uni-muenchen.de

Funding information
This work was supported by the German Federal Ministry of Education and Research (01IS18036A) and by the German Research Foundation (BO3139/4-3, BO3139/7-1, BO3139/6-2) to ALB. The authors of this work take full responsibilities for its content.

Edited by: Mehmed Kantardzic, Associate Editor and Witold Pedrycz, Editor-in-Chief

Abstract

In recent years, the need for neutral benchmark studies that focus on the comparison of methods coming from computational sciences has been increasingly recognized by the scientific community. While general advice on the design and analysis of neutral benchmark studies can be found in recent literature, a certain flexibility always exists. This includes the choice of data sets and performance measures, the handling of missing performance values, and the way the performance values are aggregated over the data sets. As a consequence of this flexibility, researchers may be concerned about how their choices affect the results or, in the worst case, may be tempted to engage in questionable research practices (e.g., the selective reporting of results or the post hoc modification of design or analysis components) to fit their expectations. To raise awareness for this issue, we use an example benchmark study to illustrate how variable benchmark results can be when all possible combinations of a range of design and analysis options are considered. We then demonstrate how the impact of each choice on the results can be assessed using multidimensional unfolding. In conclusion, based on previous literature and on our illustrative example, we claim that the multiplicity of design and analysis options combined with questionable research practices lead to biased interpretations of benchmark results and to over-optimistic conclusions. This issue should be considered by computational researchers when designing and analyzing their benchmark studies and by the scientific community in general in an effort towards more reliable benchmark results.

This article is categorized under:
Technologies > Visualization
Technologies > Data Preprocessing
Technologies > Structure Discovery and Clustering

KEYWORDS
benchmarking, method comparison, over-optimistic results, questionable research practices, variability of results

1 | INTRODUCTION AND RELATED WORK

With the constant development of new methods in computational sciences (e.g., machine learning and bioinformatics), it is becoming increasingly difficult for data analysts to keep pace with scientific progress and to select the most appropriate method for their data and research question out of the many existing approaches. This problem is addressed by benchmark studies, which systematically analyze and compare the performance of several methods in different conditions using simulated or real data sets.

In many cases, benchmark studies are performed as part of a paper introducing a new method, usually with the intention to demonstrate the superiority of the new method over existing ones. Accordingly, they can be considered as biased in favor of the newly proposed method and should be seen as an informal method comparison rather than a real benchmark study (Boulesteix et al., 2013; Buchka et al., 2021; Norel et al., 2011). In contrast, so-called *neutral* benchmark studies are defined as benchmark studies that focus on the comparison itself and are ideally performed by reasonably neutral authors, that is, authors who (1) are equally experienced with all considered methods and (2) design and analyze the study in a rational way (Boulesteix et al., 2017). These characteristics make neutral benchmark studies essentially unbiased. Therefore, recommendations resulting from such studies are especially relevant both for method users and developers (Boulesteix et al., 2018).

Regarding the appropriate design and analysis of benchmark studies, the available literature ranges from general guidelines (Boulesteix, 2015; Weber et al., 2019) and statistical frameworks (Boulesteix et al., 2015; Demšar, 2006; Eugster et al., 2012; Hothorn et al., 2005, all with focus on supervised learning), to recommendations for context-specific benchmarks (e.g., Bokulich et al., 2020; Kreutz, 2019; Mangul et al., 2019; Zimmermann, 2020). However, for many issues relevant in practice (e.g., the selection of data sets and performance measures), no concrete guidance or methodology can be found. This means that researchers are usually faced with a high amount of flexibility when conducting their benchmark study.

As a consequence, researchers who are aware of these issues, although making well-considered design and analysis choices prior to conducting the benchmark study, might be concerned about how their choices affect the results. On the other hand, the high amount of flexibility could tempt less aware researchers to engage in questionable research practices (see John et al., 2012, in the context of applied research) when conducting their benchmark study. This includes the selective reporting of results (e.g., reporting the results of only one performance measure although performance was originally assessed by two measures) and the modification of specific design and/or analysis components of the benchmark study after seeing the results (e.g., using performance measures other than those originally selected). Of course, these practices are not questionable on their own. For example, it is fine to use an alternative performance measure if the current one does not produce meaningful results as long as the change of performance measure is adequately justified and documented. However, practices such as the selective reporting of results or the post hoc modification of benchmark components do become questionable if they are applied to fit the researchers' expectations or hopes. For example, researchers might seek an "exciting" result (e.g., a clear-cut result suggesting a univocal winner as opposed to vague tendencies) or have a specific presumption in mind that they want to be confirmed by the results (e.g., the superiority of a certain method or class of methods that they are more familiar with or that has performed well in previous benchmark studies).

The problem with such research practices is that they are likely to produce over-optimistic results, that is, results with an optimistic bias towards the researchers' expectations and hopes. While we are convinced that very few researchers have the actual intention to cheat (Ioannidis et al., 2014), it should not be understated that "even an honest person is a master of self-deception" (Nuzzo, 2015), meaning that every researcher is at risk of engaging in questionable research practices. Moreover, the non-neutrality that leads to such practices in the first place is difficult to avoid completely and is likely to arise in a subconscious manner even in studies intended as neutral. Note also that the actual neutrality of neutral benchmark studies can only be checked to a certain extent. For example, one may review the authors' publication lists to identify the methods they are most familiar with, but this gives only a partial picture of someone's (non-)neutrality.

In application fields of statistics (e.g., medicine and psychology), the multiplicity of analysis strategies and the associated risk of over-optimistic results are well-known issues (Hoffmann et al., 2021; Ioannidis, 2005; Simmons et al., 2011) and terms such as "p-hacking" or "fishing expeditions" have been discussed by many (Head et al., 2015; Wagenmakers et al., 2012). However, in methodological research including benchmark studies, this topic is covered rather sparsely. Existing literature on the risk and prevention of over-optimism in benchmark studies is either limited to general considerations in benchmarking guidelines (Boulesteix et al., 2017; Weber et al., 2019) or to benchmark studies that are performed as part of a paper introducing a new method (Boulesteix, 2015; Norel et al., 2011), which can be transferred to neutral benchmark studies only to a limited extent. Similarly, the scarce literature that *empirically* investigates the effects of over-optimism in benchmark studies in a quantitative manner is either also devoted to the bias affecting evaluations of a newly proposed method to other existing methods (Buchka et al., 2021; Jelizarow et al., 2010), or focusing on the selection of data sets (MacIà et al., 2013; Yousefi et al., 2010).

In this paper, we illustrate and discuss the multiplicity of options regarding the design and analysis of neutral benchmark studies based on real data sets, and examine its effect on the results. Note that although we focus on neutral benchmark studies based on real data, our results are also relevant to benchmarks comparing new to existing methods and, to some extent, benchmarks based on simulated data. We will empirically address the multiplicity of options and its effects in a twofold approach. In the first step, in order to raise awareness of the multiplicity of possible results and the over-optimism that may arise from questionable research practices, we use the results of a recently published benchmark study to illustrate how variable the resulting method rankings are when different options for design and analysis are considered. In the second step, we propose a framework based on multidimensional unfolding (Borg & Groenen, 2005) that enables researchers to assess the impact of each choice on the method rankings. More precisely, the framework allows to analyze when and how using alternative options for a specific choice affects the results and can thus be an effective strategy to prevent biased interpretations and over-optimistic conclusions.

The exemplary study we will use throughout the paper to illustrate our proposed framework and the multiplicity of possible options and results is a benchmark experiment by Herrmann et al. (2021) comparing the performance of 13 survival prediction methods based on 18 real so-called multi-omics" data sets. Note that our paper does not intend to question the results of this study. Instead, it should be seen as extended analysis of the benchmark study, which by assessing the multiplicity of results and examining the impact of each choice, makes the results of Herrmann et al. (2021) even more reliable and meaningful.

While the framework proposed in this paper can be utilized by all researchers who conduct benchmark studies of computational methods (e.g., in the fields of machine learning, data mining, statistics, etc.), the illustrated multiplicity of results should ideally also raise awareness among the readers of such studies. The concepts and results presented in this paper may therefore be useful for method developers and methodological researchers as well as applied researchers and data analysts.

The remainder of this paper is structured as follows: we review and discuss a selection of design and analysis choices in the context of benchmark studies in Section 2, and describe the design of the study as well as the principle of multidimensional unfolding in Section 3. The results are presented in Section 4, which is followed by a discussion in Section 5 and concluding remarks in Section 6.

2 | EXAMPLES OF DESIGN AND ANALYSIS CHOICES IN BENCHMARK STUDIES

2.1 | Setting

In this section, we discuss some of the choices that researchers are faced with when conducting a benchmark study based on real data sets. In general, most choices that have to be made to conduct a benchmark study relate to (1) the general aim of the study, (2) the design of the study, or (3) the analysis of the performance results; see the left part of Figure 1. Choices that belong to the first category are, for instance, the choice of methods to be compared or the type of outcome variable to be considered. However, in this paper, we focus on choices regarding the design of the study (i.e., how the aim of the study is addressed) and the analysis of performance results (i.e., how the $L \times M$ matrix of results generated by each considered performance measure is analyzed, where L and M are the numbers of data sets and methods, respectively). It is important to note that these choices should ideally be made prior to conducting the benchmark study. However, we conjecture that they are in practice often made post hoc, that is, after seeing the

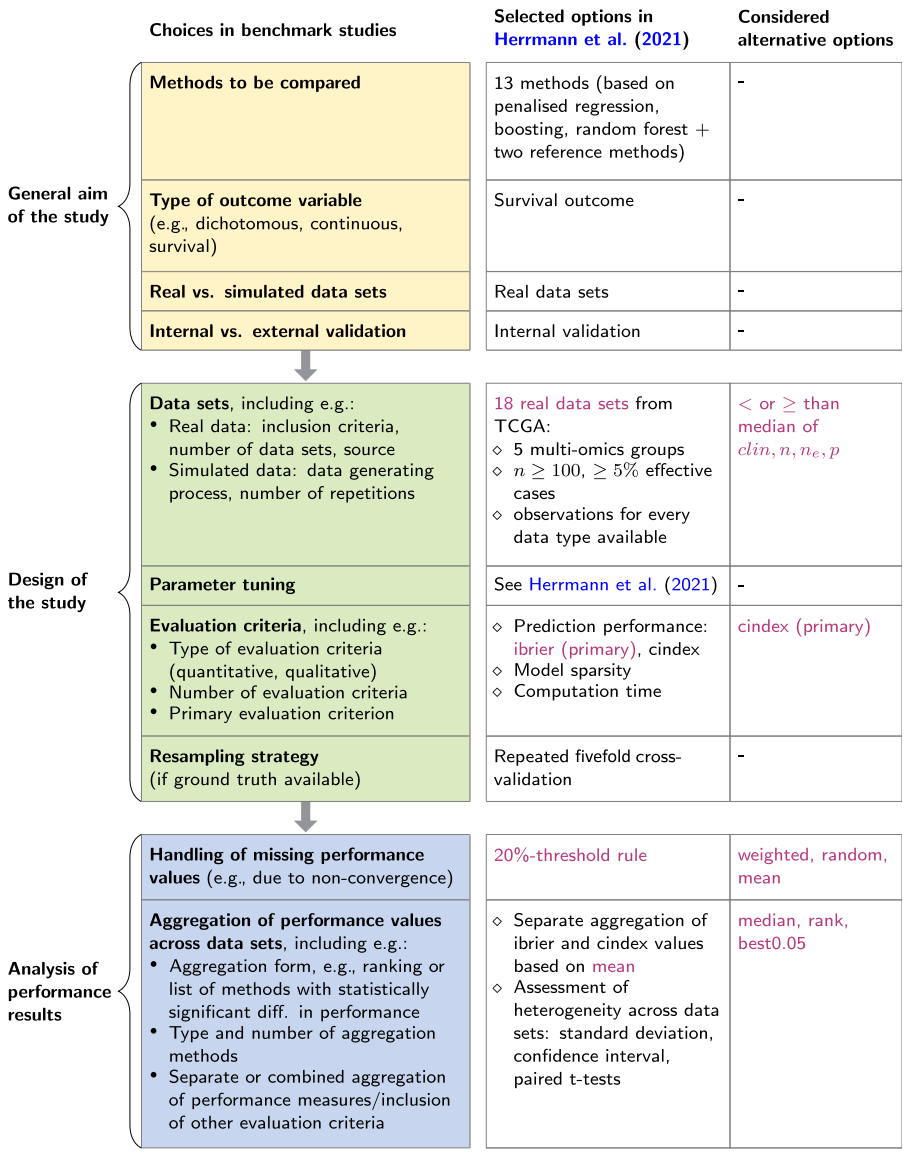

FIGURE 1 Examples of choices that researchers are usually faced with when conducting a benchmark study including options used in the example benchmark study by Herrmann et al. (2021) (second column) and alternative options (third column). Options that are considered in our illustration are colored in pink

results—which can amount to questionable research practices. When reading a benchmark study, there is no way to check when the choices were made.

For each choice, we will give concrete examples of possible options that will later be analyzed with regard to their effect on the results; see the right part of Figure 1. For this purpose, we consider the benchmark study by Herrmann et al. (2021) mentioned above. The authors compare the performance of $M = 13$ survival prediction methods (here denoted as *BlockForest*, *Clinical Only*, *CoxBoost*, *CoxBoost Favoring*, *Glmboost*, *Grridge*, *Ipflasso*, *Kaplan–Meier*, *Lasso*, *Prioritylasso*, *Prioritylasso Favoring*, *Ranger* and *Rfsrc*) on $L = 18$ real multi-omics data sets. See the original paper (Herrmann et al., 2021) for details on the methods, the benchmark experiment, and the results. We selected this study as an example because some of the authors of the present paper were also involved in conducting the benchmark study

by Herrmann et al. (2021). We therefore had first hand insight about the issues Herrmann et al. (2021) faced while designing and analyzing the benchmark study, which we believe to be reasonably representative of the important challenges encountered in most benchmark studies, as we will discuss in the remainder of this section.

2.2 | Design choices

2.2.1 | Data sets

The selection of data sets is an important design choice in every benchmark study, as the performances are usually highly variable across data sets (Novianti et al., 2015; Weber et al., 2019). To make meaningful statements and prevent the study from being underpowered, it is recommended to consider an adequate number of data sets (Boulesteix et al., 2017). Although there are suggestions on how to calculate the minimum required number (Boulesteix et al., 2015), it seems that the number of included data sets is usually based on practical criteria (such as availability or computational cost) rather than statistical considerations (MacIà et al., 2013). Moreover, if the benchmark study aims at external validation, the number of data sets that can be included in the benchmark study is usually limited, as for many data sets there is often no comparable data set available that could be used for external validation.

Concerning the type of data sets, researchers should include data sets that are representative for the domain of interest and diverse enough to make sure the methods can be evaluated under a wide range of conditions (Gatto et al., 2016; Weber et al., 2019). Corresponding inclusion criteria for the data sets should be defined before conducting the benchmark study (Boulesteix et al., 2017). However, the decision on how the inclusion criteria are defined lies with the researcher. In many benchmark studies, the exact search strategy or inclusion criteria are not reported transparently, suggesting that in these cases, there might be no clearly defined inclusion criteria at all.

In the benchmark study by Herrmann et al. (2021), the authors selected all cancer data sets with five different multi-omics groups and more than 100 samples from the TCGA research network (http://cancergenome.nih.gov). Additionally, they excluded data sets that did not have observations for every data type or less than 5% effective cases (i.e., patients with event), resulting in a total of $L = 18$ data sets. However, depending on their research interest, Herrmann et al. (2021) could have set additional constraints. For example, if the authors had been interested in the performance of the methods on data sets with a small number of effective cases, they could have adjusted the inclusion criteria accordingly (e.g., set $n_e < 30$). The other way around, one may decide to ignore data sets with a small number of events (e.g., set $n_e \geq 30$) because it is questionable if it makes sense to fit models in this case at all.

In this paper, we will address the multiplicity of possible options regarding the selection of data sets and its impact on the results by considering subgroups of the original $L = 18$ data sets defined based on some of the data sets' characteristics. The considered characteristics are the number of clinical variables (*clin*), the number of observations (n), the number of effective observations (n_e), and the number of variables (p). For each data set characteristic, we will only consider data sets that are smaller ($<$) or greater or equal ($\geq$) than the median value of the respective data set characteristic over the 18 considered data sets. This results in eight groups with 8–10 data sets.

2.2.2 | Quantitative performance measure

Another important aspect of benchmarking is the choice of evaluation criteria, which usually includes both quantitative performance measures and other measures such as runtime or qualitative features such as user-friendliness. Although all these evaluation criteria are important, we will focus on quantitative performance measures in this paper.

The choice of performance measure is usually context-specific, that is, it depends on the type of methods and data addressed in the benchmark study, as well as on the aspects of performance that are considered the most important by the researcher (Morris et al., 2019; Weber et al., 2019). It is also often a nontrivial choice. For some tasks such as classification, researchers are spoilt for choice considering the variety of measures they can choose from (e.g., accuracy, sensitivity/specificity, area under the curve or F1-score), which makes decisions difficult (Mangul et al., 2019; Robinson & Vitek, 2019). In contrast, for more complex situations they might have to design their own performance measures, which can also be challenging (Weber et al., 2019). To provide a more complete picture of the methods' behavior and avoid over-optimism, it can be useful to consider more than one performance measure (Norel et al., 2011). However,

there is no way to objectively determine the adequate number of performance measures as this is highly context dependent.

In the benchmark study by Herrmann et al. (2021), the primary performance measure is the integrated Brier score (Graf et al., 1999; denoted as *ibrier*). Additionally, they consider Uno's *C*-index (Uno et al., 2011; denoted as *cindex*). The authors justify their decision to use the ibrier as primary measure by the fact that cindex only measures the discriminatory power and is not a strictly proper scoring rule (Blanche et al., 2019), while the ibrier additionally measures calibration. However, they argue that if the main interest lies in *ranking* patients according to their risk, then the cindex would also be a valid measure. Furthermore, they reason that it makes sense to include the cindex for the purpose of comparability with other studies, since it is a widely used performance measure. Accordingly, depending on which aspect of performance they would have considered more important, Herrmann et al. (2021) could have also used the cindex as primary performance measure or only selected one of the two performance measures. In this paper, we will thus compare the results of ibrier and cindex.

2.3 | Analysis choices

2.3.1 | Handling of missing performance values

Because of non-convergence or other computational issues, methods sometimes fail to output a result for a specific data set. In the context of resampling procedures such as cross-validation or bootstrapping, the consequence is that performance values may be missing for all or part of the resampling iterations for some data sets. This problem seems to be common especially in benchmarks of larger scale (Bischl et al., 2013). While there is at least some literature devoted to the selection of data sets and performance measures, the issue of missing performance values in some combinations of data sets and methods is almost completely ignored. Many authors of benchmark studies do not report how they handled missing performance values, and there is to our knowledge no corresponding guidance available.

Bischl et al. (2013) mention several possible ad hoc options that could be applied if the missing values occur only on a subset of resampling iterations, namely that missing values could be imputed by the worst possible value or by the mean of the remaining performance values obtained for this combination of data set and method—although both options are not ideal in their opinion. Another ad hoc option they actually use for their benchmark study is a mixed strategy, where the imputed value is sampled from an estimated normal distribution of the remaining values if the method fails in less than 20% of the resampling iterations. If the method fails in more than 20% of the resampling iterations, the worst possible value is used for imputation. Herrmann et al. (2021), who use cross-validation as resampling procedure and also face the problem of failing iterations, use a similar 20%-threshold rule as Bischl et al. (2013). However, instead of sampling from a normal distribution, they use the mean performance value of the remaining iterations and instead of the worst possible value, they assign values of the performance measures corresponding to random prediction (i.e., 0.25 for ibrier and 0.5 for cindex).

Since there seems to be no common agreement on how to handle missing values in this context, other sensible options would also be justifiable. For example, missing values could be imputed by a formula that weights the mean performance value and the random performance value used by Herrmann et al. (2021) according to the proportion of missing values, thus avoiding the choice of an arbitrary threshold. For the ibrier, where 0 corresponds to the best possible value and 0.25 to random prediction, the imputed value for the considered combination of data set and method could be defined as

$$x_{\text{impute}} = 0.25 - \left(0.25 - \frac{\sum_{i \in \mathcal{I}} x_i}{|\mathcal{I}|}\right) \cdot (1 - r), \tag{1}$$

where $\mathcal{I}$ is the set of indices of the non-failed iterations, x_i is the ibrier value for iteration $i \in \mathcal{I}$, and r is the proportion of missing values. For two methods with the same mean value for non-failed iterations, the method with more missing values obtains a worse performance value. Moreover, the imputed value is equal to 0.25 if a method has 100% failures for a data set, or a mean value greater or equal than 0.25 (which makes sense since fluctuations above the value 0.25 corresponding to random prediction are not relevant). Another advantage of this weighted imputation procedure is that

it reduces to the mean when the proportion of missing values r tends to 0—as intuitively expected. The corresponding formula for the cindex can be found in the Supplementary material.

In this paper, we will consider four imputation methods that can be used to handle the issue of missing performance values: the 20%-threshold rule used by Herrmann et al. (2021), the weighted method in Equation (1), imputation using values that correspond to random prediction, and imputation using the average of the non-failed iterations.

2.3.2 | Aggregation of performance values across data sets

Although it is common to analyze the methods' individual performances across data sets (e.g., using graphical tools), most benchmark studies ultimately aggregate the performance values over the data sets to generate an overall method evaluation. This is done, for example, in the form of a ranking (often taking not only the rank order into account, but also the aggregated performance values that generate these ranks) or a list of methods that show statistically significant differences in performance. While there is much literature addressing statistical testing procedures in benchmark experiments based on a single data set (Dietterich, 1998; Hothorn et al., 2005) or several data sets (Demšar, 2006; Eisinga et al., 2017), there seems to be no consensus on how to generate an overall method *ranking* from several data sets, which we will focus on in this section.

For example, the performance values can be aggregated using standard summary measures such as the mean, median, minimum, maximum, or standard deviation (Mersmann et al., 2015). Since the distribution of performance values can be considerably skewed, some authors advise against using the mean or median as aggregation method. Instead, they recommend assigning ranks to the methods for each data set such that the best method in the corresponding data set obtains rank 1 and the worst method rank M, where M is the number of considered methods (Demšar, 2006; Hornik & Meyer, 2007). The resulting ranks are then usually aggregated using the mean (e.g., Kibekbaev & Duman, 2016; Verenich et al., 2019) or, less often, the median (e.g., Orzechowski et al., 2018).

Other possible aggregation methods include counting the number of times a method performs best, often divided by the number of data sets to obtain a value between 0 and 1 (e.g., De Cnudde et al., 2020; Fernández-Delgado et al., 2014; Wu et al., 2020). Some of these authors suggest to not only consider the best performing method for each data set but also the set of methods performing similarly to the best method. Accordingly, Fernández-Delgado et al. (2014) consider the number of data sets in which a method achieves 95% or more of the maximum accuracy (i.e., the accuracy achieved by the best performing method in that data set) divided by the total number of data sets. In the same vein, Wu et al. (2020) estimate the probability of achieving good performance as the number of data sets for which the method is among the top three methods divided by the total number of data sets.

Note that all aggregation methods presented so far are based on point estimates of the methods' performances. Although less frequently used in practice, it is also possible to generate method rankings based on the results of statistical tests (i.e., pairwise comparisons indicating if Method 1 performs significantly better than Method 2) using consensus rankings (Hornik & Meyer, 2007).

If more than one performance measure and/or other evaluation criteria (e.g., runtime) are considered, researchers also have to decide if rankings arising from multiple criteria should be combined in some form (e.g., Eugster et al., 2012) or should be considered separately, as suggested by Weber et al. (2019). Specifically, Weber et al. (2019) recommend to identify a set of consistently high performing methods based on the individual rankings and then highlight the different strengths of each method.

Herrmann et al. (2021) aggregate the performance values based on ibrier and cindex using the mean and consider each ranking separately. To assess the heterogeneity of performances across data sets, they also calculate the resulting standard deviations and confidence intervals and perform paired t-tests. In our illustration, we will consider four aggregation methods that can be used to generate method rankings: mean (as used by Herrmann et al., 2021), median, mean rank, and number of times a method performs best. If two methods obtain the same rank according to the number of times they perform best, they are additionally ranked by the number of times their performance lies within the 5% environment of the best performing method. This applies if $\frac{|\bar{x}_m - \bar{x}_{best}|}{\bar{x}_{best}} < 0.05$, where $\bar{x}_m$ denotes the performance (cindex or ibrier) of method m and $\bar{x}_{best}$ the performance of the best performing method in the corresponding data set. We denote this aggregation method (i.e., counting the number of times a method performs best and the number of times it lies within the 5% environment as secondary ranking method) as *best0.05*.

Note that we will focus on the ranks resulting from each aggregation method instead of the aggregated performance values that generate these ranks since the four aggregation methods have different scales (cindex/ibrier for mean and

median, mean ranks for mean rank and counts for best0.05), which would require appropriate normalization to compare them in a meaningful way. While this normalization would be specific to the type of considered evaluation criteria and aggregation methods, ranks can be generated in almost every benchmark study, which is why they are used in this illustrative example. Moreover, since we only evaluate the results of one performance measure at a time (ibrier or cindex), we are not considering different options for combining rankings that result from more than one performance measure.

3 | METHODS

3.1 | Design of the study

To illustrate the variability of benchmark results with respect to design and analysis choices, we use the benchmark results from Herrmann et al. (2021) and systematically examine different combinations of design and analysis options. Specifically, we consider all combinations of options regarding the choice of data sets (9 options), performance measure (2 options), imputation method (4 options), and aggregation method (4 options) described in Section 2 and Figure 1. This results in $9 \times 2 \times 4 \times 4 = 288$ combinations. We then compare the 288 resulting rankings of the 13 survival prediction methods, where a rank of 1 corresponds to the best performing method and a rank of 13 to the worst performing method (average ranks are assigned in case of ties).

3.2 | Multidimensional unfolding

The impact of each choice on the method rankings is assessed using multidimensional unfolding (Borg & Groenen, 2005; Coombs, 1964), which we will briefly introduce in the remainder of this section. Multidimensional unfolding is a technique that represents preference data as distances in a low-dimensional space. It locates K ideal points representing the subjects (in our case, $K = 288$ combinations) and M object points representing the objects (in our case, $M = 13$ methods) such that the distances from each ideal point to the object points correspond to the observed preference values. The closer an object point lies to a subject's ideal point, the stronger the subject's preference for that object. Accordingly, the ideal point itself corresponds to maximal preference (Borg et al., 2013). Note that this intuitive representation of preferences is the main reason why multidimensional unfolding is preferred over other, more widely used methods for dimension reduction, such as principal component analysis, that could alternatively be used to analyze the method rankings (for details on the differences see Chapter 16.2 in Borg & Groenen, 2005).

Multidimensional unfolding takes non-negative dissimilarities δ_{km} ($k = 1,...,K$; $m = 1,...,M$) as input, which are the preference values possibly converted in a way that small values correspond to high preferences. In our case, where the preference values are ranks, this is not necessary since a small rank already indicates high preference. Moreover, the number of dimensions dim must be specified, which we set to $dim = 2$ as it is done in most applications of multidimensional unfolding. To find the coordinates for the points representing the K subjects and M objects, a loss function (*stress*) is minimized. It is defined as

$$\sigma^2\left(\hat{\boldsymbol{D}}, \boldsymbol{Z}_1, \boldsymbol{Z}_2\right) = \sum_{k=1}^{K} \sum_{m=1}^{M} w_{km} \left(\hat{d}_{km} - d_{km}(\boldsymbol{Z}_1, \boldsymbol{Z}_2)\right)^2, \tag{2}$$

where w_{km} denotes a non-negative a priori weight (which is set to $w_{km} = 1$ by default), and $\boldsymbol{Z}_1 \in \mathbb{R}^{K \times dim}$ and $\boldsymbol{Z}_2 \in \mathbb{R}^{M \times dim}$ are the coordinates for the points representing the subjects and objects, respectively. Moreover, $d_{km}(\boldsymbol{Z}_1, \boldsymbol{Z}_2)$ denotes the fitted Euclidean distances

$$d_{km}(\boldsymbol{Z}_1, \boldsymbol{Z}_2) = \sqrt{\sum_{s=1}^{dim} (z_{1ks} - z_{2ms})^2}. \tag{3}$$

The matrix $\hat{\boldsymbol{D}} \in \mathbb{R}_0^{+K \times M}$ contains the disparities $\hat{d}_{km} = f(\delta_{km})$, which are the optimally scaled dissimilarities. This means that the loss function in Equation (2) is not only minimized with respect to $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ but also with respect to a

function $f(\cdot)$ that transforms the dissimilarities δ_{km} into disparities $\hat{d}_{km}$ (the function class depends on the assumed scale level). If, as in our example, the preference data are available in the form of ranks, $f(\delta_{km})$ reflects a monotone step function that is found through monotonic regression on the dissimilarities. This type of multidimensional unfolding is referred to as ordinal or non-metric unfolding. However, multidimensional unfolding can also be easily applied if the preference data are on a metric scale level by simply employing a different function class. In our example benchmark study, such metric preference data could be aggregated ibrier or cindex values, for instance.

To avoid degenerate solutions due to equal disparities which occur particularly often in non-metric unfolding, it is recommended to use a penalized version of the stress function in (2) that involves the coefficient of variation $v(\hat{\boldsymbol{D}})$. The penalized stress function is minimized through numerical optimization using a strategy called SMACOF (Stress Majorization of a Complicated Function) and is implemented in an R package of the same name (de Leeuw & Mair, 2009). For details on multidimensional unfolding and its implementation see Mair et al. (2021), Borg and Groenen (2005), and Busing et al. (2005).

4 | RESULTS

For full reproducibility, the entire analysis and the results presented in this section are publicly available in the GitHub repository https://github.com/NiesslC/overoptimism_benchmark.

4.1 | Overall variability and step-wise optimization

As a first step, we compare the method rankings resulting from all 288 combinations of design and analysis options. Figure 2 shows the corresponding rank distribution for each method. Importantly, it reveals that any method can achieve almost any rank. On one hand, all methods but one achieve rank 1 (8 methods) or 2 (4 methods) for at least

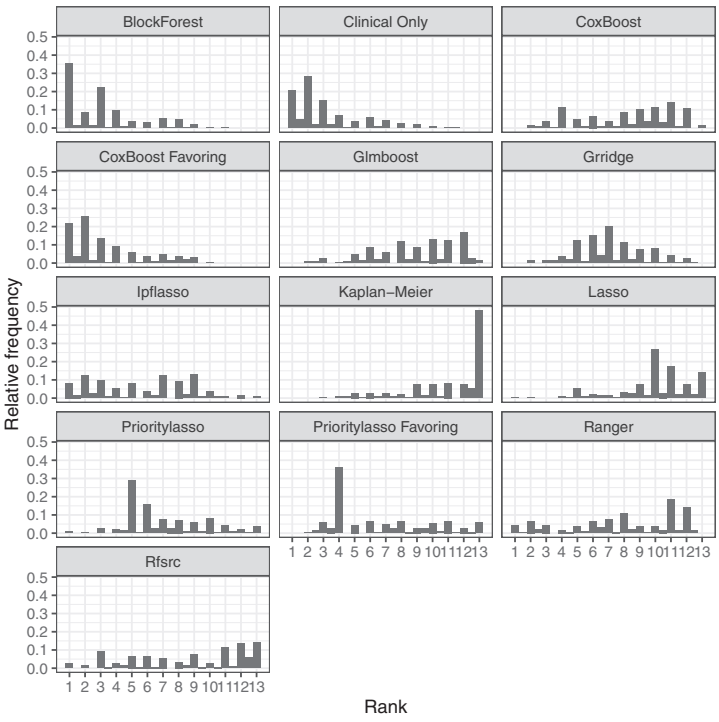

FIGURE 2 Rank distribution of 13 methods generated by 288 combinations of design and analysis options

one combination. The exception is Kaplan–Meier, which does not use any feature information and can achieve ranks as small as 3. On the other hand, 10 methods are found to be the worst or one of the two worst methods (i.e., have rank 13 or 12.5, respectively) for at least one combination. The highest rank obtained by the remaining methods (Clinical Only, BlockForest, and CoxBoost Favoring) ranges from 10 to 11.5. Figure 2 also reveals that the ranks are distributed differently for each method. For example, while Clinical Only obtains rank 1 or 2 in approximately 50% of the combinations, the ranks of Ranger are more evenly distributed.

While considering all combinations of options provides valuable information on the overall variability of results, it is not a realistic scenario concerning over-optimism in the sense that no researcher conducting a benchmark study would try all possible combinations to obtain a favorable result (unless they are actively cheating, which we do not assume here). Therefore, we additionally illustrate how easy it is to modify the method rankings if the design and analysis options are selected in a step-wise optimization process, which might represent a more realistic scenario. In our illustration, the step-wise optimization for each method is performed as follows: In each step (i.e., for each choice), the option that yields the best rank for the considered method (or the best performance value in case of equal ranks) is selected. If all options yield the same result, the default option is used. As default options, we use all 18 data sets, ibrier as primary performance measure, the 20%-threshold rule as imputation method, and the mean as aggregation method. This corresponds to the setting of Herrmann et al. (2021). Moreover, we assume that a favorable result is a small rank for a specific method. Note that this may not always be the case, for example, if one expects a reference method such as Kaplan–Meier to obtain a high rank or considers a group of several methods as target.

Figure 3 displays the optimization process if the ranks are optimized in the order: (1) imputation method, (2) aggregation method, (3) performance measure, and (4) data sets. It shows that for 8 of 13 methods, the best rank achieved by step-wise optimization corresponds to the smallest possible rank for the corresponding method (i.e., the smallest rank that can be achieved when all 288 combinations are considered) and for another three methods, the step-wise optimization achieves one rank higher than the smallest possible rank. Only two methods (Prioritylasso and Grridge) show a larger discrepancy between step-wise optimization and considering all possible combinations. However, this is not too surprising considering the few cases and thus very specific combinations where they achieve small ranks (see Figure 2). If a step is missing in the optimization process of a certain method, this indicates that the corresponding step did not

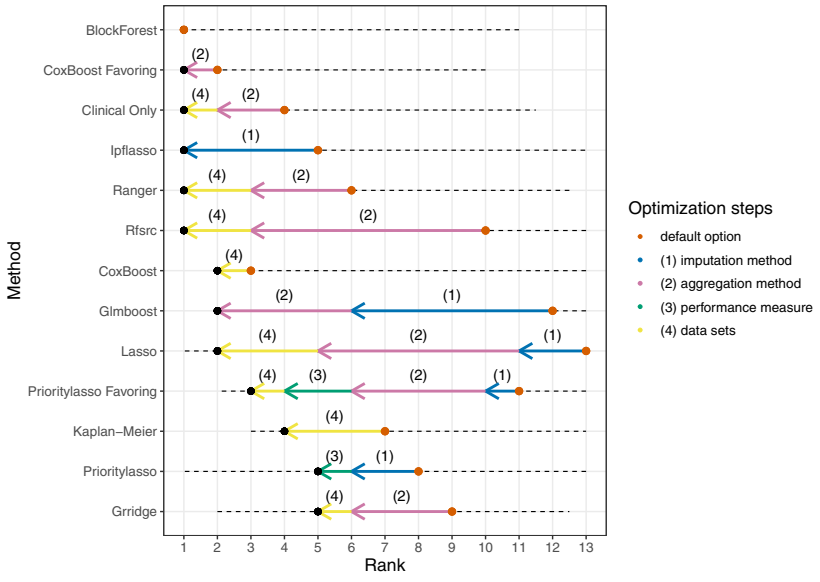

FIGURE 3 Step-wise optimization of method ranks by (1) imputation method (blue), (2) aggregation method (pink), (3) performance measure (green), and (4) data sets (yellow). The dotted line corresponds to the smallest and highest possible ranks when all 288 combinations are considered. Missing steps indicate that they did not lead to an improved rank. Default options correspond to Herrmann et al. (2021)

improve the rank of that method. In fact, all methods except Lasso and Prioritylasso Favoring require no more than two optimization steps.

Note that the results of the step-wise optimization depend on the default options. For example, when cindex instead of ibrier is used as default option, the resulting ranks are higher (see Figure S1). Moreover, the results depend on the order in which the ranks are optimized. The order shown in Figure 3 is realistic in the sense that researchers might find it more problematic to modify components of the benchmark study that are generally considered as important (i.e., performance measure or data sets) and thus only resort to them if the previous optimization steps (i.e., imputation method or aggregation method) do not yield a favorable result. However, other orders in which the ranks are optimized would also be conceivable. For example, the selection of data sets could be optimized first since it offers many options and can be easily modified by eliminating specific data sets. In this case, the selection of data sets remains the only optimization step for many methods since the subsequent steps do not lead to an improvement (see Figure S2), which already indicates the large impact of data set selection, discussed in more detail in the next section.

4.2 | Impact of individual design and analysis choices

To gain additional insight concerning the impact of each design and analysis choice, the method rankings are analyzed using multidimensional unfolding. Figure 4 displays the resulting unfolding solution that represents the rankings of all 288 combinations regarding the 13 methods. Before looking at the different colorings of the ideal points in Figure 4a–d, we can make some general observations on how the combinations and methods are scaled in the plot (which is identical for each figure). First, the unfolding solution clearly shows that the method rankings can differ widely depending on which combination of design and analysis options is considered, which is consistent with the results presented in Section 4.1. Second, similar to the rank distribution in Figure 2, the unfolding solution indicates that some methods tend to achieve smaller ranks than other methods. This applies specifically to Clinical Only, CoxBoost Favoring, and BlockForest, which are scaled close to the origin and thus have a small distance to most ideal points. In contrast, other methods such as Lasso and Kaplan–Meier can be found in the periphery of the plot, indicating that they obtain rather high ranks by most combinations.

Of course, the degree to which the presented unfolding solution reflects the actual rankings depends on its goodness-of-fit (a perfect fit usually requires as many dimensions as there are methods, i.e., $dim = M = 13$). However, following Mair et al. (2016), the unfolding solution in Figure 4 fits the ranking data reasonably well (see the Supplementary material for diagnostic figures and measures).

An important feature of the unfolding solution in Figure 4 is that not only the distances between ideal and object points can be interpreted, but also the distances within ideal and object points. This means that, in contrast to the rank distribution in Figure 2, the unfolding solution also provides information about which methods are ranked similarly and which combinations of design and analysis options yield similar rankings. We make use of the latter (i.e., the fact that the unfolding solution indicates which combinations yield similar rankings) to assess the impact of each design and analysis choice on the method rankings. For this purpose, the unfolding solution is supplemented with additional information, which results in Figure 4a–d: For each choice, the ideal points are colored according to the option that was used in the respective combination, with the default option (i.e., the option used in Herrmann et al., 2021) colored in gray. Moreover, we connect each ideal point representing the default option to the ideal points representing the alternative options given that the other three choices remain the same. Although this makes the representation dependent on which option is used as the default, for reasons of clarity, we refrain from additionally connecting the alternative options with each other.

The resulting plot for the choice of performance measure is displayed in Figure 4a. The gray lines indicate that the distances between most ideal points corresponding to pairs of ibrier and cindex within one specific setting (i.e., combinations where the other three choices remain the same) are large. Accordingly, the choice of performance measure strongly impacts the resulting method ranking for most settings. Figure 4a also reveals that the ideal points corresponding to ibrier and cindex form two clearly separated clusters. Accordingly, the variability in the method rankings is reduced if the performance measure is fixed. This applies in particular to the cindex, whose corresponding ideal points show considerably less variation than the ideal points corresponding to the ibrier. With regard to the remaining

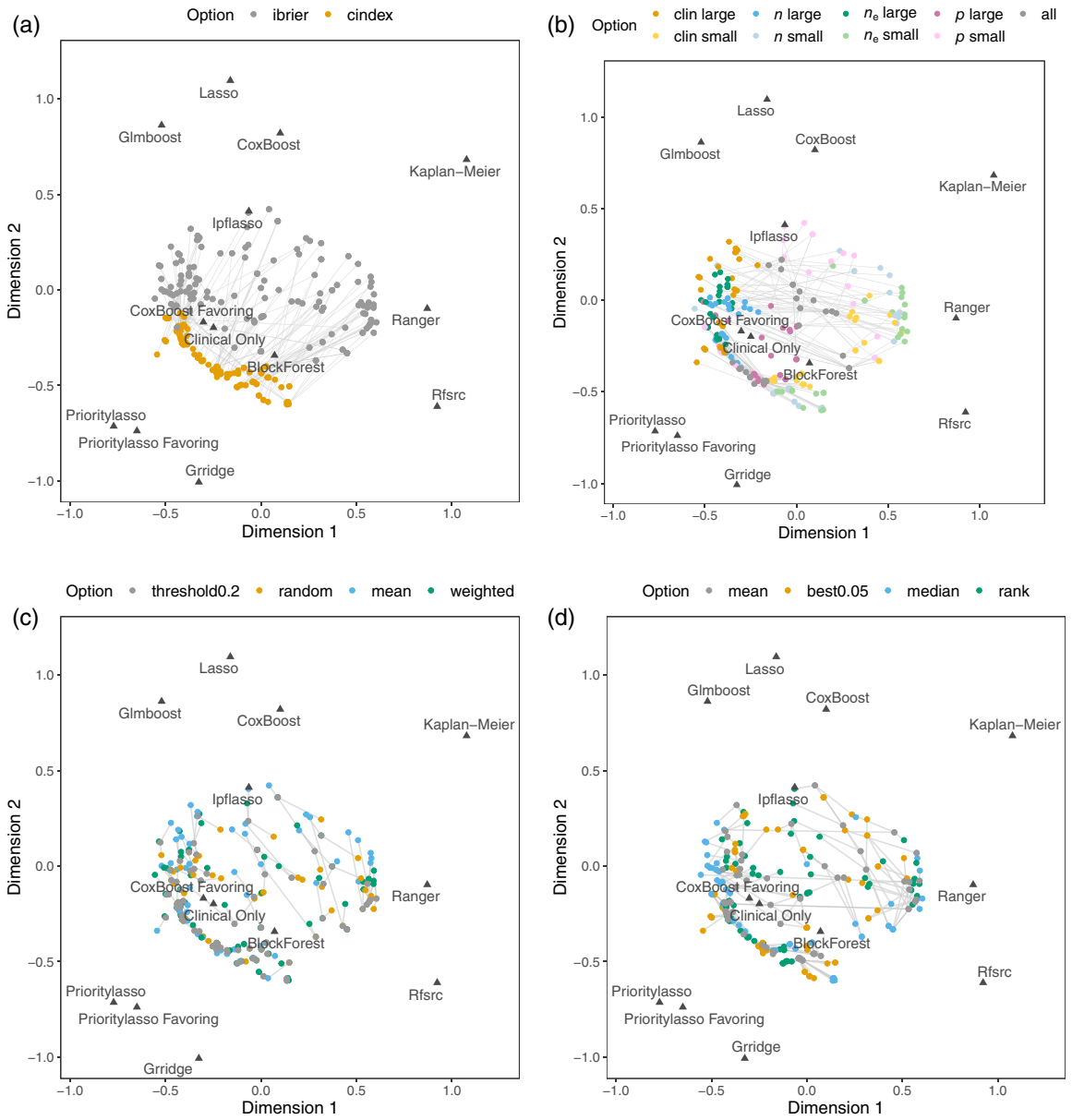

FIGURE 4 Unfolding solution representing the rankings of 288 combinations of design and analysis options (*ideal points*; circles) regarding 13 methods (triangles). For each choice, the ideal points are colored according to the option that was used in the respective combination (default options corresponding to Herrmann et al., 2021 are gray). Each ideal point representing a default option is connected to the ideal points representing alternative options, given that the other three choices remain the same. (a) Performance measure, (b) data sets, (c) imputation method, and (d) aggregation method

three choices (data sets, imputation method, and aggregation method), this means that their impact is smaller if the cindex is used as performance measure. This finding might be explained by the fact that the cindex only measures discriminatory power (see Section 2) and might thus be more robust to changes in the remaining design and analysis choices than the ibrier.

As can be seen from Figure 4b, another important choice that accounts for a large part of the variability in the method rankings is the selection of data sets, especially if the ibrier is used as performance measure (compare with Figure 4a). Figure 4b also reveals that within the two clusters corresponding to cindex and ibrier, the ideal points are roughly clustered according to the group of data sets that was used in the respective combination. This indicates that keeping the data sets fixed in addition to the performance measure again reduces the variability in the method rankings. Regarding the type of data sets used in each combination, Figure 4b shows that within both clusters of performance measure, the ideal points corresponding to small and large values of each data set characteristic lie approximately opposite to each other while the ideal points representing all 18 data sets are located between them. With regard to the choice of data sets, the largest discrepancy between two rankings can thus be expected when comparing the results of two groups that correspond to small and large values of one of the considered data set characteristics. Using all 18 data sets, on the other hand, results in a compromise between the two extremes.

As already stated above, the variability in the method rankings is considerably reduced if performance measure and data sets are fixed, which in turn means that the variations caused by using different imputation or aggregation methods are expected to be small. This finding is confirmed by Figure 4c,d. The gray lines indicate that variations in the method rankings caused by deviations from the default imputation or aggregation method mainly arise for ibrier as the performance measure and all groups of data sets except those with many clinical variables or large values of n or n_e (compare with Figure 4a,b). In some of the other settings, the impact of the choice of imputation and aggregation method is so small that the ideal points corresponding to different imputation/aggregation methods have the same coordinates (i.e., yield the same ranking). This applies in particular to the choice of imputation method, which generally has less impact on the method rankings than the choice of aggregation method, as can be seen from comparing Figure 4c and Figure 4d.

The distances between ideal points of default and alternative options that are represented as gray lines in Figure 4a–d can also be summarized as boxplots, which are displayed in Figure 5. This representation provides information that is technically also included in Figure 4a–d, but is presented more clearly in Figure 5. For example, it shows for each choice which alternative option used instead of the default option tends to yield the highest variations in the method rankings (e.g., for the choice of imputation method, it is the option that uses the mean of the non-failed iterations as imputation value). Moreover, Figure 5 reveals that according to the unfolding solution, the largest discrepancy between two rankings generated by only varying one design or analysis option is achieved by using the median instead of the mean as aggregation method. This is an unexpected finding since it has already been stated above and can also be seen from Figure 5 that in most settings (i.e., combinations where the other three choices remain the same), the choice of aggregation method tends to have a smaller impact on the method rankings than the choice of performance

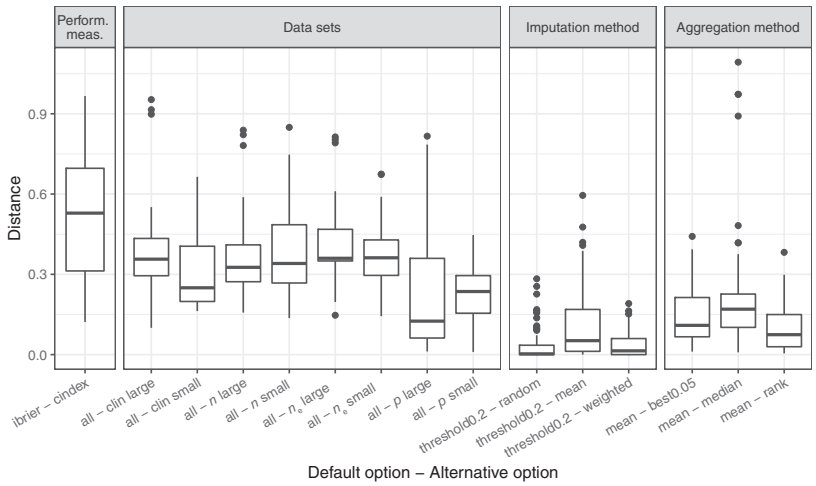

FIGURE 5 Distances between ideal points of combinations that represent default and alternative options of one specific choice (given that the other three choices remain the same), derived from the unfolding solution in Figure 4. The larger the distance, the larger the discrepancy between the two rankings generated by using the alternative option instead of the default option

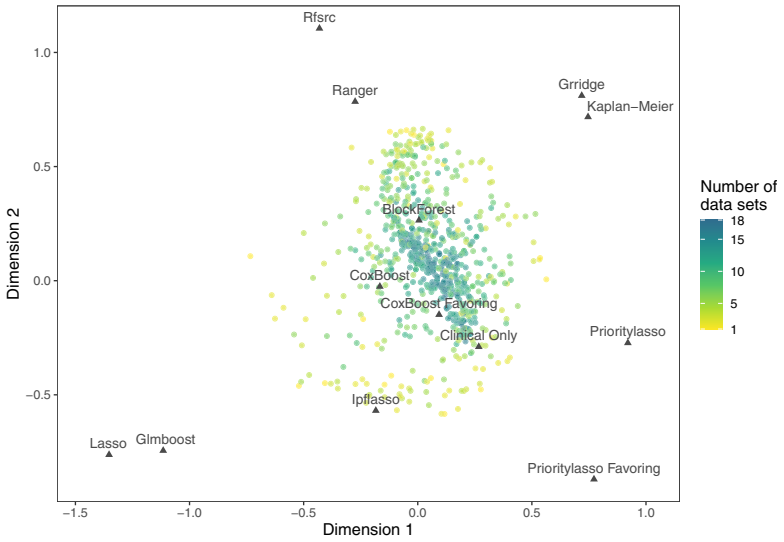

FIGURE 6 Unfolding solution representing 774 rankings (circles) of 13 methods (triangles) generated by randomly sampling different groups of data sets while performance measure, imputation method, and aggregation method are fixed to their respective default option

measure and data sets. A major drawback of Figure 5 is that in contrast to Figure 4a–d, it does not provide any information about how similar the rankings generated by the alternative options are, nor about how the ranks of the individual methods change.

Of course, all findings concerning the impact of the individual design and analysis choices depend on the number and type of options considered for each choice. Specifically, for the choice of data sets, we only consider a small subset of possible options and we focus, in addition to the 18 original data sets, on groups of approximately equal size (8–10 data sets) generated by specific data set characteristics. We thus complement our analysis by illustrating the impact of the choice of data sets if more options are considered, especially with regard to the number of data sets. For this purpose, we keep performance measure, imputation method, and aggregation method fixed to their respective default option and randomly draw 50 permutations of the 18 original data sets. For each of these permutations we store the method rankings generated by only considering the first l data sets with $l = 1, ..., 17$, and remove duplicate groups of data sets (which mainly occur for groups with 1, 2, or 17 data sets). This results in 774 rankings including one ranking generated by the 18 original data sets, which are all represented in the unfolding solution in Figure 6. The widely distributed ideal points clearly indicate that the choice of data sets is even more essential if the number of data sets is not restricted and the groups of data sets are not defined based on specific data set characteristics (as it was the case above in Figure 4). As one might have expected, we also observe that the variability in the method rankings increases if the number of data sets decreases. Accordingly, the most extreme rankings (i.e., rankings that differ the most from the ranking generated using all 18 data sets) occur for groups with only a few data sets. Since Figure 4a revealed that the impact of the choice of data sets strongly depends on the choice of performance measure, we repeat the analysis using cindex as performance measure (see Figure S3). Similar to Figure 4b, the impact of the choice of data sets is considerably reduced. However, as in Figure 6, the variability in the method rankings increases with decreasing number of data sets.

5 | DISCUSSION

5.1 | Summary

In this paper, we addressed the multiplicity of design and analysis options in the context of benchmark studies and the associated risk of over-optimistic results. As a preliminary step, we reviewed literature related to the choice of four design and analysis choices that researchers are usually faced with when conducting a benchmark study based on real

data sets, namely the choice of data sets, the choice of quantitative performance measure, the choice of imputation method for missing performance values, and the choice of aggregation method to generate an overall method ranking.

We then used the benchmark study by Herrmann et al. (2021) to illustrate how variable the resulting method rankings of a benchmark study can be when all possible combinations of a range of design and analysis options are considered. In fact, in this example, the results were so variable that any method could achieve almost any rank, that is, each method could almost be presented as best or worst method for at least one combination of design and analysis options. For the more realistic scenario where the design and analysis options are not systematically examined for each combination but selected in a step-wise optimization process, we observed that the variability in the method rankings is smaller but still remarkable.

In addition to examining the overall variability in the method rankings, we also investigated the individual impact of each choice on the results using multidimensional unfolding. As might be expected, the choice of performance measure and data sets accounts for a large part of the variability in the method rankings. The impact of the choice of imputation and aggregation method, on the other hand, tends to be considerably smaller but still non-negligible in many settings. In general, the impact of each choice depends on the options used for the other three choices, with the choice of performance measure affecting the impact of the remaining choices most strongly. In an additional analysis, we increased the number of considered options for the choice of data sets, which clearly showed that the variability in the method rankings increases if the number of data sets decreases and once again emphasized the importance of the choice of data sets.

5.2 | Limitations

Of course, the specific results obtained for the example study by Herrmann et al. (2021) should only be seen as an illustration that cannot be generalized to other benchmark studies. Moreover, one possible reason why the method rankings are so variable is that in our example benchmark study, many performance differences are small and the performance values differ widely across data sets, as discussed in the original study by Herrmann et al. (2021). The focus of our study was on ranks, which do not reflect the size of the differences between the methods' performances or the heterogeneity across data sets. On the one hand, taking these aspects into account rather than focusing on ranks may lead to much less variable results, particularly if one relies on statistical tests. On the other hand, the multiplicity of possible analysis options is not limited to the analysis of ranks: there are also plenty of possibly ways to analyze performance differences and the heterogeneity across data sets, even if statistical tests are performed (e.g., paired *t*-test or Wilcoxon signed-rank test with or without correction for multiple testing, or global tests such as the Friedman test).

5.3 | Negative consequences and possible solutions

Despite these limitations, our illustration suggests that, as a consequence of the multiplicity of design and analysis options, the results of benchmark studies could be much more variable than many researchers realize. Combined with questionable research practices (e.g., the selective reporting of results or the targeted modification of specific design and analysis components), this potentially high variability of benchmark results can lead to biased interpretations and over-optimistic conclusions regarding the performance of some of the considered methods. Given the high level of evidence that is attributed to neutral benchmark studies (Boulesteix et al., 2017), a "neutral" benchmark study that is in fact biased could thus negatively affect both methodological and applied research by misleading method users and developers (Weber et al., 2019).

Fortunately, there are several strategies to prevent over-optimistic benchmark results that arise from the multiplicity of design and analysis options, some of which are already applied by many researchers, including Herrmann et al. (2021). For example, strategies inspired from blinding in clinical trials can help to reduce non-neutrality and/or the potential to exploit the multiplicity of possible options. Specifically, blinding could be realized by labeling the methods with non-informative names (e.g., Method A, Method B, etc.) such that researchers have no information about the performance of each method until the end of the study (Boulesteix et al., 2017). If the benchmark study is based on simulated data, researchers could also be blinded to the data generation process, which prohibits the possibility to tune the parameters of selected methods according to the known ground truth (e.g., Kreutz et al., 2020).

The remaining strategies to prevent over-optimistic results can be summarized using the work of Hoffmann et al. (2021), who formalize the effect of both random sources of uncertainty (including sampling uncertainty) and

epistemic sources of uncertainty (resulting in a multiplicity of possible analysis strategies and thus opening the door to questionable research practices) on the replicability of research findings. They outline six steps researchers from all empirical research fields can take to make their own research more replicable and credible. In brief, researchers should (1) be aware of the multiplicity of possible analysis strategies, (2) reduce uncertainty, (3) integrate uncertainty, (4) report uncertainty, (5) acknowledge uncertainty, and (6) publish all research code, data and material. Although Hoffmann et al. (2021) focus on applied rather than methodological research, we argue that their recommended steps can also be applied to address the sources of uncertainty that arise from the design and analysis of benchmark studies.

Step 1. In the context of benchmark studies, the first step to reduce the risk of over-optimistic results is to simply be aware of the multiplicity of possible design and analysis options and the potential for questionable research practices. We can only speculate about how much awareness for this issue is already present in methodological research but hope that this paper contributes to raising it.
Step 2. The second step suggested by Hoffmann et al. (2021) is to reduce sources of uncertainty. In the context of benchmark studies, this could be realized by consulting existing benchmarking guidelines found in literature. However, as discussed in this paper, guidelines for many issues relevant in practice are still lacking. We claim that more guidance and standardized approaches are needed in this context. Regarding the choice of data sets, uncertainty could be reduced if the number of data sets to include in the study would be consequently based on statistical considerations such as power calculation (e.g., Boulesteix et al., 2015) and if data sets would be selected according to strict and well-considered inclusion criteria. Both aspects are facilitated if structured and well-documented databases exist for the type of data to be studied.
Step 3. As a third step, Hoffmann et al. (2021) recommend to integrate remaining sources of uncertainty that could not be reduced in the second step. Analysis approaches such as confidence intervals, statistical tests, or boxplots that take the heterogeneity of performance values across data sets into account can be seen as first steps towards integrating the uncertainty regarding the choice of data sets. However, they do not provide much information about how the benchmark results would change if only certain subgroups of data sets would be considered. A more advanced but less common way to integrate uncertainty regarding the choice of data sets is to analyze the relationship between method performance and data set characteristics (e.g., Eugster et al., 2014; Kreutz et al., 2020; Oreski et al., 2017). Concerning the choice of evaluation criteria (including quantitative performance measures), the aggregation of method rankings resulting from different criteria into an overall ranking can be seen as an attempt towards integrating uncertainty. However, to our knowledge, currently existing approaches such as consensus rankings (Hornik & Meyer, 2007) do not provide any measure of uncertainty.
Step 4. For all sources that cannot be adequately integrated, Hoffmann et al. (2021) suggest to systematically report the results of alternative analysis strategies, which, in the context of benchmark studies, would be alternative design and analysis options. While reporting the results of alternative analysis strategies, for example, in the form of a sensitivity analysis, is a common procedure in applied research (Hoffmann et al., 2021), to our knowledge it is rarely performed in benchmark studies (especially if they are based on real data sets). However, considering the lack of ways to reduce and integrate uncertainty when designing and analyzing benchmark studies, adequately reporting the results of alternative options seems to be all the more important. One reason for the lack of uncertainty reporting in benchmark studies could be that, to our knowledge, no suitable framework has been available so far. This gap could be filled by the framework based on multidimensional unfolding that we used in this paper. It can be seen as a systematic version of standard sensitivity analysis that allows to graphically assess the variability of the method rankings with respect to a large number of different combinations of design and analysis options. It also provides information about the individual impact of each choice on the method ranking and thus enables researchers to analyze when and how using alternative options for a specific choice affects the results. In this way, the risk of misleading readers is reduced and the benchmark results become even more reliable and valuable. Moreover, using the framework allows to identify critical choices that substantially affect the results and should therefore be particularly well justified in future benchmark studies and be given more consideration in benchmarking guidelines.
Step 5. The next important step suggested by Hoffmann et al. (2021) is to accept the inherent uncertainty of scientific findings. In the context of benchmark studies, this implies that researchers should clearly state that the benchmark results are conditional on the selected design and analysis options (Boulesteix et al., 2013; Hornik & Meyer, 2007). In this vein, researchers should also acknowledge that just as in applied research, generalizations from a single study are usually not appropriate (Amrhein et al., 2019; Hoffmann et al., 2021). This emphasizes the need for more high-quality benchmark studies and for meta-analyses of benchmark studies (e.g., Gardner et al., 2019), which, however, are still rare and unfortunately sometimes not considered as full-fledged research by the scientific community (Boulesteix

et al., 2020). Another aspect also related to the acceptance of uncertainty is to recognize that statistical inference within exploratory analyses should be treated with great caution (Amrhein et al., 2019; Hoffmann et al., 2021). Similar to applied research, strictly confirmatory benchmark studies could be realized by pre-registration of design- and analysis plans, as recently implemented in the context of the so-called pre-registration experiment (see https://preregister.science) or through the registered report" publication format (Chambers, 2013), which has meanwhile been adopted by several interdisciplinary journals that also accept computational papers. It is also important to recall that there is usually no best method for all scenarios and data sets (the well-known "no free lunch" theorem; Wolpert, 2002). Especially for data sets and evaluation criteria, it might thus be advisable to accept the uncertainty that is associated with their choice by putting more focus on the analysis of the individual strengths and weaknesses of each method than on an aggregated overall ranking. This can for example be realized by individually analyzing the rankings generated by each evaluation criterion and by investigating the relationship between method performance and data set characteristics (see Step 3).

Step 6. As a final step, the publication of codes and (if possible) data sets that ideally allow the extension to alternative options and additional methods can reduce the impact of over-optimism since it enables readers to run alternative analyses and to reveal potentially biased results.

The strategies provided in this section are also summarized in a checklist (Table S1), which can assist researchers when designing and analyzing benchmark studies.

6 | CONCLUSION

In conclusion, our illustration suggests that benchmark results can be highly variable with respect to design and analysis choices, which can lead to biased interpretations and over-optimistic conclusions. However, there is a wide range of strategies that can help to avoid these pitfalls. We hope that our proposed framework makes a useful contribution towards this objective. While a certain amount of over-optimism can probably never be completely avoided, addressing this problem will lead to more reliable and valuable benchmark results.

ACKNOWLEDGMENT

The authors thank Anna Jacob for language correction.

CONFLICT OF INTEREST

The authors have declared no conflicts of interest for this article.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in GitHub at https://github.com/NiesslC/overoptimism_benchmark.

AUTHOR CONTRIBUTIONS

Christina Nießl: Conceptualization (equal); formal analysis (lead); methodology (lead); visualization (lead). **Moritz Herrmann:** Conceptualization (equal); data curation (equal); methodology (supporting). **Chiara Wiedemann:** Conceptualization (equal); data curation (equal); methodology (supporting). **Giuseppe Casalicchio:** Conceptualization (equal); methodology (supporting). **Anne-Laure Boulesteix:** Conceptualization (equal); funding acquisition (equal); methodology (supporting); supervision (equal).

ORCID

Christina Nießl https://orcid.org/0000-0003-2425-7858
Moritz Herrmann https://orcid.org/0000-0002-4893-5812
Giuseppe Casalicchio https://orcid.org/0000-0001-5324-5966
Anne-Laure Boulesteix https://orcid.org/0000-0002-2729-0947

RELATED WIREs ARTICLE

Method evaluation, parameterization, and result validation in unsupervised data mining: A critical survey

REFERENCES

Amrhein, V., Trafimow, D., & Greenland, S. (2019). Inferential statistics as descriptive statistics: There is no replication crisis if we don't expect replication. *The American Statistician*, *73*, 262–270.

Bischl, B., Schiffner, J., & Weihs, C. (2013). Benchmarking local classification methods. *Computational Statistics*, *28*, 2599–2619.

Blanche, P., Kattan, M. W., & Gerds, T. A. (2019). The *c*-index is not proper for the evaluation of *t*-year predicted risks. *Biostatistics*, *20*, 347–357.

Bokulich, N. A., Ziemski, M., Robeson, M. S., & Kaehler, B. D. (2020). Measuring the microbiome: Best practices for developing and benchmarking microbiomics methods. *Computational and Structural Biotechnology Journal*, *18*, 4048–4062.

Borg, I., & Groenen, P. J. F. (2005). *Modern multidimensional scaling: Theory and applications* (2nd ed.). Springer.

Borg, I., Groenen, P. J. F., & Mair, P. (2013). *Applied multidimensional scaling*. Springer.

Boulesteix, A.-L. (2015). Ten simple rules for reducing overoptimistic reporting in methodological computational research. *PLoS Computational Biology*, *11*, e1004191.

Boulesteix, A.-L., Binder, H., Abrahamowicz, M., & Sauerbrei, W. (2018). On the necessity and design of studies comparing statistical methods. *Biometrical Journal*, *60*, 216–218.

Boulesteix, A.-L., Hable, R., Lauer, S., & Eugster, M. J. A. (2015). A statistical framework for hypothesis testing in real data comparison studies. *The American Statistician*, *69*, 201–212.

Boulesteix, A.-L., Hoffmann, S., Charlton, A., & Seibold, H. (2020). A replication crisis in methodological research? *Significance*, *17*, 18–21.

Boulesteix, A.-L., Lauer, S., & Eugster, M. J. A. (2013). A plea for neutral comparison studies in computational sciences. *PLoS One*, *8*, e61562.

Boulesteix, A.-L., Wilson, R., & Hapfelmeier, A. (2017). Towards evidence-based computational statistics: Lessons from clinical research on the role and design of real-data benchmark studies. *BMC Medical Research Methodology*, *17*, 138.

Buchka, S., Hapfelmeier, A., Gardner, P. P., Wilson, R., & Boulesteix, A.-L. (2021). On the optimistic performance evaluation of newly introduced bioinformatic methods. *Genome Biology*, *22*, 152.

Busing, F. M. T. A., Groenen, P. J. K., & Heiser, W. J. (2005). Avoiding degeneracy in multidimensional unfolding by penalizing on the coefficient of variation. *Psychometrika*, *70*, 71–98.

Chambers, C. D. (2013). Registered reports: A new publishing initiative at Cortex. *Cortex*, *49*, 609–610.

Coombs, C. H. (1964). *A theory of data*. Wiley.

De Cnudde, S., Martens, D., Evgeniou, T., & Provost, F. (2020). A benchmarking study of classification techniques for behavioral data. *International Journal of Data Science and Analytics*, *9*, 131–173.

de Leeuw, J., & Mair, P. (2009). Multidimensional scaling using majorization: SMACOF in R. *Journal of Statistical Software*, *31*, 1–30.

Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*, 1895–1923.

Eisinga, R., Heskes, T., Pelzer, B., & Grotenhuis, M. (2017). Exact *p*-values for pairwise comparison of Friedman rank sums, with application to comparing classifiers. *BMC Bioinformatics*, *18*, 68.

Eugster, M. J. A., Hothorn, T., & Leisch, F. (2012). Domain-based benchmark experiments: Exploratory and inferential analysis. *Austrian Journal of Statistics*, *41*, 5–26.

Eugster, M. J. A., Leisch, F., & Strobl, C. (2014). (Psycho-)analysis of benchmark experiments: A formal framework for investigating the relationship between data sets and learning algorithms. *Computational Statistics and Data Analysis*, *71*, 986–1000.

Fernández-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, *15*, 3133–3181.

Gardner, P. P., Watson, R. J., Morgan, X. C., Draper, J. L., Finn, R. D., Morales, S. E., & Stott, M. B. (2019). Identifying accurate metagenome and amplicon software via a metaanalysis of sequence to taxonomy benchmarking studies. *PeerJ*, *7*, e6160.

Gatto, L., Hansen, K. D., Hoopmann, M. R., Hermjakob, H., Kohlbacher, O., & Beyer, A. (2016). Testing and validation of computational methods for mass spectrometry. *Journal of Proteome Research*, *15*, 809–814.

Graf, E., Schmoor, C., Sauerbrei, W., & Schumacher, M. (1999). Assessment and comparison of prognostic classification schemes for survival data. *Statistics in Medicine*, *18*, 2529–2545.

Head, M. L., Holman, L., Lanfear, R., Kahn, A. T., & Jennions, M. D. (2015). The extent and consequences of p-hacking in science. *PLoS Biology*, *13*, e1002106.

Herrmann, M., Probst, P., Hornung, R., Jurinovic, V., & Boulesteix, A.-L. (2021). Large scale benchmark study of survival prediction methods using multi-omics data. *Briefings in Bioinformatics*, *22*, bbaa167. https://doi.org/10.1093/bib/bbaa167

Hoffmann, S., Schönbrodt, F., Elsas, R., Wilson, R., Strasser, U., & Boulesteix, A.-L. (2021). The multiplicity of analysis strategies jeopardizes replicability: Lessons learned across disciplines. *Royal Society Open Science*, *8*, 201925.

Hornik, K., & Meyer, D. (2007). Deriving consensus rankings from benchmarking experiments. In R. Decker & H.-J. Lenz (Eds.), *Advances in data analysis* (pp. 163–170). Springer.

Hothorn, T., Leisch, F., Zeileis, A., & Hornik, K. (2005). The design and analysis of benchmark experiments. *Journal of Computational and Graphical Statistics*, *14*, 675–699.

Ioannidis, J. P. A. (2005). Why most published research findings are false. *PLoS Medicine*, *2*, e124.

Ioannidis, J. P. A., Munafò, M. R., Fusar-Poli, P., Nosek, B. A., & David, S. P. (2014). Publication and other reporting biases in cognitive sciences: Detection, prevalence, and prevention. *Trends in Cognitive Sciences*, *18*, 235–241.

Jelizarow, M., Guillemot, V., Tenenhaus, A., Strimmer, K., & Boulesteix, A.-L. (2010). Overoptimism in bioinformatics: An illustration. *Bioinformatics*, *26*, 1990–1998.

John, L. K., Loewenstein, G., & Prelec, D. (2012). Measuring the prevalence of questionable research practices with incentives for truth telling. *Psychological Science*, *23*, 524–532.

Kibekbaev, A., & Duman, E. (2016). Benchmarking regression algorithms for income prediction modeling. *Information Systems*, *61*, 40–52.

Kreutz, C. (2019). Guidelines for benchmarking of optimization approaches for fitting mathematical models. *Genome Biology*, *20*, 281.

Kreutz, C., Can, N. S., Bruening, R. S., Meyberg, R., Mérai, Z., Fernandez-Pozo, N., & Rensing, S. A. (2020). A blind and independent benchmark study for detecting differentially methylated regions in plants. *Bioinformatics*, *36*, 3314–3321.

MacIà, N., Bernadó-Mansilla, E., Orriols-Puig, A., & Kam Ho, T. (2013). Learner excellence biased by data set selection: A case for data characterisation and artificial data sets. *Pattern Recognition*, *46*, 1054–1066.

Mair, P., Borg, I., & Rusch, T. (2016). Goodness-of-fit assessment in multidimensional scaling and unfolding. *Multivariate Behavioral Research*, *51*, 772–789.

Mair, P., Groenen, P. J. F., & de Leeuw, J. (2021). More on multidimensional scaling and unfolding in R: smacof version 2. *Journal of Statistical Software*. https://cran.r-project.org/web/packages/smacof/vignettes/smacof.pdf

Mangul, S., Martin, L. S., Hill, B. L., Lam, A. K. M., Distler, M. G., Zelikovsky, A., Eskin, E., & Flint, J. (2019). Systematic benchmarking of omics computational tools. *Nature Communications*, *10*, 1393.

Mersmann, O., Preuss, M., Trautmann, H., Bischl, B., & Weihs, C. (2015). Analyzing the BBOB results by means of benchmarking concepts. *Evolutionary Computation*, *23*, 161–185.

Morris, T. P., White, I. R., & Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, *38*, 2074–2102.

Norel, R., Rice, J. J., & Stolovitzky, G. (2011). The self-assessment trap: Can we all be better than average? *Molecular Systems Biology*, *7*, 537.

Novianti, P. W., Jong, V. L., Roes, K. C., & Eijkemans, M. J. (2015). Factors affecting the accuracy of a class prediction model in gene expression data. *BMC Bioinformatics*, *16*, 199.

Nuzzo, R. (2015). How scientists fool themselves —And how they can stop. *Nature*, *526*, 182–185.

Oreski, D., Oreski, S., & Klicek, B. (2017). Effects of dataset characteristics on the performance of feature selection techniques. *Applied Soft Computing*, *52*, 109–119.

Orzechowski, P., La Cava, W., & Moore, J. H. (2018). Where are we now? A large benchmark study of recent symbolic regression methods. In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '18*, Association for Computing Machinery, New York, NY, USA (pp. 1183–1190).

Robinson, M. D., & Vitek, O. (2019). Benchmarking comes of age. *Genome Biology*, *20*, 205.

Simmons, J. P., Nelson, L. D., & Simonsohn, U. (2011). False-positive psychology: Undisclosed flexibility in data collection and analysis allows presenting anything as significant. *Psychological Science*, *22*, 1359–1366.

Uno, H., Cai, T., Pencina, M. J., D'Agostino, R. B., & Wei, L. J. (2011). On the *C*-statistics for evaluating overall adequacy of risk prediction procedures with censored survival data. *Statistics in Medicine*, *30*, 1105–1117.

Verenich, I., Dumas, M., La Rosa, M., Maggi, F. M., & Teinemaa, I. (2019). Survey and cross-benchmark comparison of remaining time prediction methods in business process monitoring. *ACM Transactions on Intelligent Systems and Technology*, *10*, 34.

Wagenmakers, E. J., Wetzels, R., Borsboom, D., van der Maas, H. L. J., & Kievit, R. A. (2012). An agenda for purely confirmatory research. *Perspectives on Psychological Science*, *7*, 632–638.

Weber, L. M., Saelens, W., Cannoodt, R., Soneson, C., Hapfelmeier, A., Gardner, P. P., Boulesteix, A.-L., Saeys, Y., & Robinson, M. D. (2019). Essential guidelines for computational method benchmarking. *Genome Biology*, *20*, 125.

Wolpert, D. H. (2002). The supervised learning no-free-lunch theorems. In R. Roy, M. Köppen, S. Ovaska, T. Furuhashi, & F. Hoffmann (Eds.), *Soft computing and industry: Recent applications* (pp. 25–42). Springer.

Wu, Z., Zhu, M., Kang, Y., Leung, E. L.-h., Lei, T., Shen, C., Jiang, D., Wang, Z., Cao, D., & Hou, T. (2020). Do we need different machine learning algorithms for QSAR modeling? A comprehensive assessment of 16 machine learning algorithms on 14 QSAR data sets. *Briefings in Bioinformatics*, *22*, bbaa321. https://doi.org/10.1093/bib/bbaa321

Yousefi, M. R., Hua, J., Sima, C., & Dougherty, E. R. (2010). Reporting bias when using real data sets to analyze classification performance. *Bioinformatics*, *26*, 68–76.

Zimmermann, A. (2020). Method evaluation, parameterization, and result validation in unsupervised data mining: A critical survey. *WIREs Data Mining and Knowledge Discovery*, *10*, e1330.

SUPPORTING INFORMATION

Additional supporting information may be found in the online version of the article at the publisher's website.

How to cite this article: Nießl, C., Herrmann, M., Wiedemann, C., Casalicchio, G., & Boulesteix, A.-L. (2022). Over-optimism in benchmark studies and the multiplicity of design and analysis options when interpreting their results. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *12*(2), e1441. https://doi.org/10.1002/widm.1441

Part III.

Unsupervised Manifold Learning

7. Unsupervised Functional Data Analysis via Nonlinear Dimension Reduction

Chapter 7 focuses on hyperparameter tuning of unsupervised manifold learning methods. Extensive experiments on simulated functional data are conducted to assess a tuning approach based on ranking-based embedding quality criteria. Particular attention is given to the effect of phase variation in functional data. The insights gained are illustrated by three real-world examples.

Contributing article:

Herrmann, M., & Scheipl, F. (2020). Unsupervised Functional Data Analysis via Nonlinear Dimension Reduction. arXiv preprint arXiv:2012.11987. https://arxiv.org/abs/2012.11987

Copyright information:

Author contributions:

Moritz Herrmann had the idea of dealing with the topic in this way and wrote the paper. Fabian Scheipl made a substantial contribution by continuously revising the manuscript and adding ideas.

Supplementary material available at:

Code and data: https://github.com/HerrMo/fda-ndr

UNSUPERVISED FUNCTIONAL DATA ANALYSIS VIA NONLINEAR DIMENSION REDUCTION

PREPRINT

Moritz Herrmann*
moritz.herrmann@stat.uni-muenchen.de
Department of Statistics
Ludwig-Maximilians-University
Munich, Germany

Fabian Scheipl
fabian.scheipl@stat.uni-muenchen.de
Department of Statistics
Ludwig-Maximilians-University
Munich, Germany

ABSTRACT

In recent years, manifold methods have moved into focus as tools for dimension reduction. Assuming that the high-dimensional data actually lie on or close to a low-dimensional nonlinear manifold, these methods have shown convincing results in several settings. This manifold assumption is often reasonable for functional data, i.e., data representing continuously observed functions, as well. However, the performance of manifold methods recently proposed for tabular or image data has not been systematically assessed in the case of functional data yet. Moreover, it is unclear how to evaluate the quality of learned embeddings that do not yield invertible mappings, since the reconstruction error cannot be used as a performance measure for such representations. In this work, we describe and investigate the specific challenges for nonlinear dimension reduction posed by the functional data setting. The contributions of the paper are three-fold: First of all, we define a theoretical framework which allows to systematically assess specific challenges that arise in the functional data context, transfer several nonlinear dimension reduction methods for tabular and image data to functional data, and show that manifold methods can be used successfully in this setting. Secondly, we subject performance assessment and tuning strategies to a thorough and sys-

*Corresponding author

tematic evaluation based on several different functional data settings and point out some previously undescribed weaknesses and pitfalls which can jeopardize reliable judgment of embedding quality. Thirdly, we propose a nuanced approach to make trustworthy decisions for or against competing nonconforming embeddings more objectively.

Keywords Dimension reduction · Functional data analysis · Manifold methods · Unsupervised learning.

1 Introduction

The ever-growing amount of easily available high-dimensional data has led to an increasing interest in methods for dimension reduction in several contexts, for example, image processing [13] and single cell data [2, 16]. Next to standard dimension reduction methods such as Principal Component Analysis (PCA) and Multidimensional Scaling (MDS), manifold methods have moved into focus in recent years. If the assumption that high-dimensional data actually lie on or close to a lower-dimensional Riemannian manifold holds, i.e., if the data have low *intrinsic* dimension, nonlinear dimension reduction methods are often capable of detecting this intrinsic low-dimensional structure even if standard, in particular linear, methods fail to do so. In this paper, we describe and assess a general approach for extending established and state of the art manifold methods ISOMAP [37], DIFFMAP [9], t-SNE [28], and UMAP [29] to functional data and use MDS as a default reference method for benchmarking.

Functional data analysis (FDA) [34, 41, e.g.], which is an active field of research in statistics with many close connections to time series analysis, focuses on data in which the units of observation are realizations of stochastic processes over compact domains. This kind of data is another data type for which the manifold assumption is often reasonable: On the one hand, such data is infinite dimensional in theory and typically very high-dimensional in practice – functional observations are usually recorded on fine and dense grids: For example, spectroscopic measurements are typically evaluated on thousands of electromagnetic wavelengths or electrocardiograms, measured at 100 Hz for 10 minutes, would yield 60,000 grid points each. On the other hand, such signals typically contain a lot of structure, and it is often reasonable to assume that only a few modes of variation suffice to describe most of the information contained in the data, i.e., such functional data often have low intrinsic dimension, at least approximately.

An important complication is that FDA often faces the challenge of two kinds of variation, both of which can be of major interest: amplitude (i.e., "vertical") variation affecting the slope, level, and size of local extrema of a function and phase (i.e., "horizontal") variation affecting the location of extrema and inflection points. Phase variation, which can be conceptualized as elastic deformations of the domain of the functional observations, often results in complex nonlinear intrinsic structure [7]. As our results show, this is true even for fairly simple phase variation structures. Despite some prior work [7, 10] showing that manifold methods for functional data – specifically, functional versions of ISOMAP [37] – can successfully deal with structured phase variation and yield efficient and compact low-dimensional representations and despite recent substantial progress in the development and application of manifold methods to tabular, image and video data [16, 42, e.g.], manifold learning for functional data remains an underdeveloped topic. However, low-dimensional representations of functional data are highly relevant for real-world problems. Finding reliable low-dimensional – especially 2- or 3-dimensional – representations of data is beneficial for visualization, description,

and exploration purposes in general. In FDA settings, this is especially crucial as the visualization of large data sets of functional observations is particularly challenging and quickly overwhelms analysts with ostensible complexity even if the underlying structures are actually fairly low-dimensional and simple, c.f Figure 2. Moreover, finding informative low-dimensional representations of functional data is an essential preprocessing step for functional data, since these representations can be used as feature vectors in supervised machine learning algorithms which require tabular, not functional data inputs [32].
In this work, we thoroughly assess if manifold methods can be used to embed functional data, perform a careful evaluation of hyper-parameter tuning approaches for functional manifold methods and investigate the suitability of the derived embeddings in various settings. Specifically, we address the following questions:
(1) Are the manifold methods under investigation able to detect low-dimensional manifold structure of functional data? Special attention is given to assessing the effects of phase variation.
(2) To what extent can automatic tuning strategies replace laborious and subjective visual inspection in order to obtain reliable embeddings in unsupervised FDA settings?

The remainder of the paper is structured as follows: In section 2 we specify notation and the theoretical framework, give a description of the embedding methods used, and an overview of performance assessment and tuning approaches. Moreover, we motivate our study design. In section 3 we describe the design of the synthetic data simulations and assess the tuning and embedding approaches in these settings, in which the "ground truth" is available for verification. The concepts and insights developed on synthetic data are then brought to bear on three real data sets in section 4. The findings of the study are finally discussed in section 5.

2 Background

2.1 Problem specification and framework

Nonlinear dimension reduction [(NDR) 3, 23, e.g.] is based on the assumption that high-dimensional data observed in a D-dimensional space $\mathcal{H}$ actually lie on or close to a d-dimensional manifold $\mathcal{M} \subset \mathcal{H}$, with $d < D$ [6]. One is then interested in finding an embedding $e : \mathcal{H} \to \mathcal{Y}$ from the high-dimensional space to a low-dimensional embedding space $\mathcal{Y}$ such that $\mathcal{Y}$ is as similar to $\mathcal{M}$ as possible.

In most NDR applications, one simply considers $\mathcal{H} = \mathbb{R}^D$. In a functional data setting, the situation is more involved: We define a d-dimensional *parameter space* $\Theta \subset \mathbb{R}^d$ while $\mathcal{F} = \mathcal{L}^2(\mathcal{T})$, the space of square integrable functions over the domain $\mathcal{T}$, takes on the role of $\mathcal{H}$, and $\phi : \Theta \to \mathcal{F}$ is a mapping from the parameter space to the *function space*. We then observe functions $x_i(t) \in \mathcal{F}$ with $x_i(t) = \phi(\theta_i)$, which can have a complex but intrinsic low dimensional structure, depending on both the structure and dimensionality of Θ and the complexity of ϕ.

Transferring this to the NDR terminology, $\mathcal{M} = \Theta$, i.e. the low-dimensional manifold is the parameter space. However, the observed data are functions in the subspace $\mathcal{M}_{\mathcal{F}} \subset \mathcal{F}$, i.e., using the terms of [7], a functional manifold. Thus, using manifold methods, an embedding $e : \mathcal{M}_{\mathcal{F}} \to \mathcal{Y}$ can be constructed. Specifically, that means we have the mappings $\Theta \xrightarrow{\phi} \mathcal{M}_{\mathcal{F}} \xrightarrow{e} \mathcal{Y}$, but only $e : \mathcal{F} \to \mathcal{Y}$ can be learned from the data

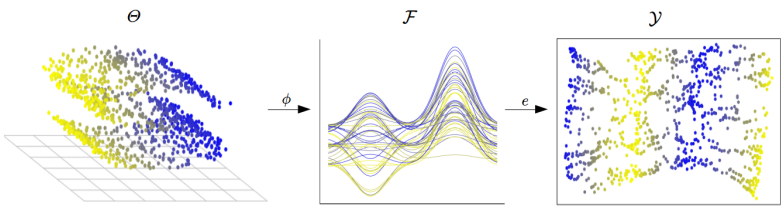

Figure 1: Framework for nonlinear dimension reduction of functional data.

(s. Figure 1). The question we try to answer is this: how well can the underlying global structure of Θ be recovered in an embedding learned from data on $\mathcal{M}_{\mathcal{F}}$?

In general, however, it is not straightforward to define what "recovering well" is supposed to mean in a specific NDR setting [12], a rarely discussed but crucial aspect. In particular, the fact that manifolds which are locally homeomorphic to $\mathbb{R}^d$ by definition need not be homeomorphic to $\mathbb{R}^d$ globally needs to be taken into account. E.g., a 2-sphere, although locally homeomorphic to the plane $\mathbb{R}^2$, can not be embedded into $\mathbb{R}^2$ globally by a single embedding e without distortions. In differential geometry terms, only specific manifolds $\mathcal{M}$ like the famous "Swiss roll" can be represented by a single chart. However, since learning an embedding function is roughly equivalent to estimating a chart of the data manifold, manifold learning faces an ill-posed problem if the atlas of a data manifold requires more than one chart.

Since it is not known in practice whether a single chart is sufficient or not, assessment of the embeddings achieved by manifold methods must always be considered under both local and global perspectives: (1) successful on a local level if the embedding $e : \mathcal{H} \to \mathbb{R}^d$ yields an embedding space $\mathcal{Y}$ in which local structures are similar to local structures in $\mathcal{M} \subset \mathcal{H}$, i.e., if e preserves neighborhoods of (small) membership size g, and (2) globally successful if $\mathcal{Y}$ is as close to $\mathcal{M}$ as possible, e.g., if an underlying "Swiss roll" manifold is "unrolled" into a plane. Note that the latter is not possible for every manifold. Thus, we consider a learned embedding to be successful if the resulting configurations of data units in $\mathcal{Y}$ are as similar to the corresponding configurations in Θ as possible in the following sense: In the conducted simulation study, we use simple parameter manifolds Θ which are mostly homeomorphic to $\mathbb{R}^d$ with $d \in \{1, 2\}$, and – in one case – homeomorphic to the circle. This allows us to evaluate embedding methods and tuning approaches for the functional data generated from Θ from a *global* perspective, both by visual inspection and quantitatively. *Local* characteristics are additionally investigated for the real data sets.

As phase variation typically transforms the domain non-linearly, phase-varying functional data is very likely to live on a non-simple functional manifold $\mathcal{M}_{\mathcal{F}}$ that is no longer globally isometric to $\mathbb{R}^d$ even if the generating parameter manifold Θ is a simple linear subspace, an issue leading to additional complexity in the FDA setting. By restricting our simulation study in part to fairly simple, linear Θ, we are able to assess these non-obvious and previously undescribed effects of domain warping on the embeddings.

2.2 Assessing performance of manifold methods

Since the methods we employ do not yield invertible embeddings e, we cannot evaluate them based on their reconstruction error $E[L(x, e^{-1}(y))]$, where L is some loss function measuring the divergence between x and its reconstruction $e^{-1}(y)$, which would objectively quantify the quality of an embedding in terms of the fidelity of its low-dimensional compression to the original data. Assessing embeddings qualitatively by visual inspection, as is widely done [e.g. 1, 6, 29], cannot be automated, and so it does not scale to large-scale comparison and benchmark studies nor to the important task of tuning a method's hyperparameters.

Instead, several surrogate measures have been developed, often based on comparing the ranks of pairwise distances between the high-dimensional data space and the learned embedding space [8, 24, 25, 39, e.g.]. We employ measures based on the normalized local continuity meta-criterion (LCMC) [22], since they are parameter free, yield a single scalar value – a property particularly desirable if the measure is supposed to be used for tuning – and allow to assess local and global performance. The LCMC is based on the measure

$$Q_{NX}(g) = \underbrace{\frac{1}{g}\frac{1}{n}\sum_{i=1}^{n} |\mathcal{N}_g^{\mathcal{H}}(i) \cap \mathcal{N}_g^{\mathcal{Y}}(i)|}_{N_g},$$

which quantifies the amount of overlap between the memberships of neighborhoods of a certain size in the two spaces [19, 24]. The g-neighborhood $\mathcal{N}_g(i)$ is defined as the set of those g objects which are closest to i in the respective space according to a suitable distance measure, and N_g measures the mean overlap obtained by averaging the cardinalities of the intersections between all such neighborhoods in the high dimensional space $\mathcal{H}$ and the low dimensional embedding space $\mathcal{Y}$. The factor $\frac{1}{g}$ is a normalization factor. The normalized LCMC is then defined as

$$R_{NX}(g) = \frac{(n-1)Q_{NX}(g) - g}{n - 1 - g},$$

which also accounts for random overlap [8, 22]. A value of 0 is expected for a random embedding, i.e., the agreement between g-neighborhoods in $\mathcal{Y}$ and in $\mathcal{H}$ is the same as that of a random configuration of objects in $\mathcal{Y}$. A value of 1 indicates a perfect embedding with complete identity of all g-neighborhoods in the two spaces [22, 24].

The choice of neighborhood size g, however, is crucial and has a strong influence on whether an embedding is judged to be successful or not. For large g, these metrics quantify the preservation of global structure in the embedding, for small g, the preservation of local structure.

To circumvent the problems that come with the choice of g, one can compute parameter free measures based on $R_{NX}(g)$ [19]. Regarding $R_{NX}(g)$ as a function of g and averaging the function values on either side of its maximum at g_{max}, leads to both a local and a global performance measure:

$$\mathrm{Q}_{\text{local}} = \frac{1}{g_{max}} \sum_{g=1}^{g_{max}} Q_{NX}(g) \qquad \text{and} \qquad \mathrm{Q}_{\text{global}} = \frac{1}{n - g_{max}} \sum_{g=g_{max}}^{n-1} Q_{NX}(g).$$

To quantify the overall performance of an embedding, the *area under the $R_{NX}(g)$-curve*

$$AUC_{R_{NX}} = \frac{\sum_{g=1}^{n-2} R_{NX}(g)}{\sum_{g=1}^{n-2} \frac{1}{g}}$$

can be computed [19]. Given such a scalar measure of performance, embedding methods can then be tuned by maximizing this measure over the different hyperparameter settings.

However, while it is known that the choice of the distance metric has a strong influence on embedding methods [40], the influence of the distance metric used to compute the g-neighborhoods in the surrogate performance measures remains unclear. Since these measures are based on g-neighborhoods, the proximity metric used to define the neighborhoods in the respective spaces is crucial, especially if the goal of the analysis is to "unfold" the global structure of the manifold. In order to recover the global manifold structure, neighborhoods in the high-dimensional space should be defined using *geodesic* distances rather than Euclidean distances, since only the former represent long-range distances on the manifold correctly, while the latter are merely distances in the ambient space. This distinction is likely to be highly relevant especially if the observed high-dimensional data manifold has a complex structure, such as $\mathcal{M}_{\mathcal{F}}$ in our situation, and when the measures are supposed to be used for automatic parameter tuning for optimal recovery of global structure. In the following, we use $AUC^m_{R_{NX}}$ and $\text{Q}^m_{\text{local}}$, where $m \in \{dir, geo\}$ indicates the distance metric used to calculate the neighborhoods for performance assessment, i.e. in this study g-neighborhoods in $R_{NX}(g)$ are computed either using L_2 distances (i.e. Euclidean distances in $\mathbb{R}^D$) or geodesic distances. In the following, we use the term *direct* distance instead of L_2 distance to emphasize the conceptual difference of distance measures which merely quantify proximity in the ambient space (hence *direct* distance) and *geodesic* distance measures quantifying proximity on a (nonlinear) manifold. Note, this is a general conceptual difference. Most standard distance metrics can be regarded as direct distance measures in the particular space and geodesic distances as computed here can be obtained based on several of these direct distance metrics. We will show that direct distances such as the L_2 distance can yield very misleading results when used in the surrogate performance measures and that tuning approaches can thus lead to far from optimal configurations.

2.3 Embedding methods and tuning approach

In this study, we compare nonlinear dimension reduction methods *isometric feature mapping* (ISOMAP) [37], *diffusion map* (DIFFMAP) [9], *t-distributed stochastic neighborhood embedding* (t-SNE) [28], and *uniform manifold approximation and unfolding* (UMAP) [29] for functional data. All these methods have *locality parameters* that control whether (rather) local structures or (rather) global structures are considered, that is how much "context" of the respective data points is taken into account while constructing the embedding. These parameters influence the result strongly and need to be tuned. We apply MDS as a simple tuning-free benchmark reference method.

ISOMAP is based on classical MDS. In contrast to MDS it is capable of unfolding the intrinsic structure of a data set. The algorithm consists of three steps. First, a nearest-neighbor graph is constructed based on a suitable direct distance metric, usually the L_2 metric. This requires defining a neighborhood size either by a distance threshold ϵ or by the number of neighbors k to be included. This parameter is the main tuning parameter of the algorithm. In the next step, shortest-path or geodesic distances among

all points are computed based on the nearest-neighbor graph. These distances are then supplied to classical MDS, which embeds the observations accordingly. ISOMAP is supposed to be particularly suited to detect global structures [37].
DIFFMAP is another spectral embedding method projecting on the eigenvectors of a diffusion operator on the data manifold. Proximity of data points is defined by a kernel function whose width acts as a tunable locality parameter [9, 27].
t-SNE, which has been state-of-the-art for several years [29], builds upon stochastic neighborhood embedding (SNE). In contrast to the aforementioned methods, (t-)SNE transforms proximities between data points into conditional probabilities of them being neighbors in the respective space and then minimizes the Kullback–Leibler divergence of the implied distribution in the original space from that in the embedding space. The perplexity of the implied distribution in the original space acts as a tunable locality parameter.
UMAP [29] is a state-of-the-art manifold learning method based on three assumptions – uniformly distributed data on a locally connected manifold equipped with a locally constant metric. It computes a fuzzy topological representation of the manifold based on a nearest-neighbor graph. The number of nearest neighbors serves as a tunable locality parameter.

In addition to the investigation of how successfully the intrinsic manifold structure of a functional data set can be detected and unfolded in general, we also want to investigate how reliably automatic tuning approaches identify suitable hyper-parameter settings. We consider the parameters steering the degree of locality as the main tuning parameter of these embedding methods. Using the terminology of the respective R packages, these are the neighborhood sizes `k` for ISOMAP, `n_neighbors` for UMAP, the `perplexity` for t-SNE and `eps.val` for DIFFMAP. Hereinafter we refer to these parameters as *locality parameters*. Moreover, the methods are not supplied with the raw data matrices, but with distance matrices instead. Note, this means that an initialization via PCA is not performed for t-SNE using `Rtsne`. To investigate these aspects, we compute both the direct and geodesic distance matrix for each simulated data set in the function space as well as in the parameter space. For a given parameter configuration, the direct distance matrix obtained from the function space is then input to the respective embedding method. Finally, direct distances of the learned embeddings are computed. As the performance measures are based on the comparison of g-neighborhoods in the high dimensional and the embedding spaces, we compute the performance measures with respect to both the parameter space as well as the function space based on direct and the geodesic distance neighborhoods in the simulation settings. Recall, direct distances represent distances in the ambient space rather than distances on the manifold and the resulting neighborhoods are thus unlikely to be well suited for performance assessment and tuning if the intrinsic structure is nonlinear, especially for larger neighborhood sizes.
Parameters are tuned for optimal performance via an extensive grid search, c.f. Table 1. For DIFFMAP we compute a dataset-specific starting value ϵ_s via `epsilonCompute`, which is the default value of the method, and use this to define the search grid of the locality parameter. In the synthetic data settings of Section 3, we can perform a "ground truth"-based meta assessment, in which we evaluate the effect of direct and geodesic distances. Moreover, we can compare the results achieved by tuning based on performance measures computed in the function space with those achieved by a practically infeasible "oracle" tuning method that uses corresponding performance measures computed in the unobservable true parameter space instead.

Table 1: Parameters of the manifold methods which are subjected to tuning. The second column shows the total amount of different parameter configurations in the tuning parameter grid, the third column the locality parameter, the fifth column the embedding dimension parameter, and the last column further parameters tuned. The "grid"-columns display the search grids of the parameters in the preceding column. The embedding dimension grid for t-SNE differs from the other grids because the implementation does not allow the embedding dimension to be greater than three.

Method	#	locality param.	grid	embedding dim.	grid	further param.
MDS	4	-	-	`k`	[2, 5]	-
ISOMAP	1300	`k`	[3, 975] step size: 3	`ndim`	[2, 5]	-
DIFFMAP	6000	`eps.val`	$[0.15\epsilon_s, 1.85\epsilon_s]$ length: 250	`neigen`	[2, 5]	`t`

UMAP	18720	`n_neighbors`	[5, 975]
			step size: 5
				`n_components`	[2, 5]	`min_dist`
						`n_epochs`
						`init`

t-SNE	21184	`perplexity`	[3, 333]
			step size: 1
				`dims`	[2, 3]	`theta`
						`max_iter`
						`eta`
						`exaggeration`

2.4 Study design

Since we are interested in whether manifold methods and tuning approaches can be used for functional data sets in general, a thorough evaluation design is inevitable. For supervised learning algorithms a wide and comprehensive body of literature exists on the conduction of neutral and objective comparison and benchmark studies [5, 11, 38, e.g.].

How to reliably evaluate algorithms and meta-learning approaches in unsupervised settings, however, is not as clear. Due to the lack of an outcome variable, clearly defined objectives to optimize against are usually not available. This makes the comparison of unsupervised learning algorithms in general, and the assessment of meta-learning approaches such as tuning in particular, prone to overoptimistic findings. General frameworks for systematic benchmarks in this context are still in their infancy [38].

In particular, nonlinear dimension reduction and manifold learning are often confronted with the lack of a clearly defined objective in terms of the achieved reconstruction error if the methods do not yield an invertible mapping. Since there is no standard benchmark procedure for unsupervised learning generally agreed upon, we devised the following procedure in order to avoid overoptimistic conclusions in our study as much as possible:

Based on the problem specification and the theoretical framework defined in Section 2.1, we first conduct a simulation study to assess embedding methods and the considered tuning approaches in settings where the ground truth is known. This allows us to investigate possible strengths, weaknesses, and pitfalls in settings that allow for objective evaluations based on a known "ground truth". We then apply the approach to real data sets where qualified assumptions about the intrinsic structure of the data can be made due to substantial considerations and analysis of previous studies. Although knowledge of the intrinsic structure is less certain than in the ground truth simulations, it is still possible to "objectively" evaluate the embeddings, at least conditional on certain substantially justified assumptions, in these settings. To some extent, this lets us examine whether the insights obtained in simulated data settings also hold for real

data. Finally, we compute embeddings for a real data set for which information about its intrinsic structure can not be justifiably inferred from prior substantial considerations – i.e., a fully unsupervised problem. We will show that such a setting can pose severe problems due to nonconforming embeddings that are hard to tackle, but that "principled" choices between nonconforming embeddings may nevertheless be possible based on the insights gained from the simulation study and from data applications in which some prior knowledge is available.

3 Simulation study

Based on the framework described in Section 2.1 we can systematically assess the utility of the described manifold methods for functional data settings by means of a simulation study. This allows to comparing the embeddings to a ground truth which is essential in an unsupervised learning problem to come to reliable conclusions. In particular, we can assess the influence of some factors likely to lead to strongly nonlinear intrinsic structure of functional data manifolds, i.e., nonlinear domain warping (phase variation) and an underlying nonlinear parameter space Θ, by systematically controlling these sources of variation.

3.1 Experiment design

Loosely speaking, the simulation design is based on two peaked functions derived from Gaussian pdfs over domain $[0, 1]$. Variation is achieved by randomly changing the locations, widths, and heights of the peaks, in total leading to eleven considered settings, six based on a linear parameter space, including three with nonlinear phase variation, and five based on a nonlinear parameter space and amplitude variation only. We can thus assess the effect of nonlinear phase variation and a nonlinear parameter space separately using the first six settings and the last five settings, respectively.

Specifically, we consider the following functional manifold

$$\mathcal{M}_{\mathcal{F}} = \left\{x \in \mathcal{L}^2([0,1]) : x(t) = \phi(\theta)\right\},$$

with $\theta = (\mathbf{a}, \mathbf{p})$ and $\phi(\theta) = b(w(t; \mathbf{p}); \mathbf{a})$. Amplitude variation in $\mathcal{M}_{\mathcal{F}}$ is parameterized as

$$b(t; \mathbf{a}) = \frac{a_1}{\sqrt{0.1\pi}}\{a_2 \cdot n(t, 0.25) + a_3 \cdot n(t, 0.75)\} + a_4,$$

with $n(t, \mu) = \exp\left(-\frac{(t-\mu)^2}{0.1}\right)$. Depending on the setting, phase variation is parameterized as the identity warping $w(t; \mathbf{p}) = t$, a linear warping

$$w(t; \mathbf{p}) = \begin{cases} p_1 t & \text{for } t \in [0, 0.5] \\ (2 - p_1)(t - 1) + 1 & \text{for } t \in (0.5, 1] \end{cases},$$

a power warping $w(t; \mathbf{p}) = t^{p_2}$ or as $w(t, \mathbf{p}) = \mathrm{B}(t; p_3, p_4)$, where $B(\cdot; a, b)$ is the cdf of a Beta(a, b) distribution.

The considered settings are obtained by selecting up to three of the parameters $a_1, a_2, a_3, a_4, p_1, p_2, p_3, p_4$, i.e., the considered settings have at most 3 degrees of freedom (df). Inactive parameters are set to constant values, e.g., $a_1 = 1$ and $a_4 = 0$.

Table 2: Overview of simulation settings.

setting	df	parameter space	variation	parameter	warping
a1-l	1	linear	amplitude	a_1	identity
p1-l	1	linear	phase	p_1	linear
c1-l	1	linear	coupled	$a_1 = p_2$	power
a2-l	2	linear	amplitude	a_2, a_3	identity
p2-l	2	linear	phase	p_3, p_4	beta cdf
i2-l	2	linear	independent	a_1, p_2	power
a2-sr	2	1D Swiss roll	amplitude	a_2, a_3	identity
a3-hx	3	1D helix	amplitude	a_2, a_3, a_4	identity
a3-sr	3	2D Swiss roll	amplitude	a_2, a_3, a_4	identity
a3-sc	3	2D S-curve	amplitude	a_2, a_3, a_4	identity
a3-tp	3	2D tp surface	amplitude	a_2, a_3, a_4	identity

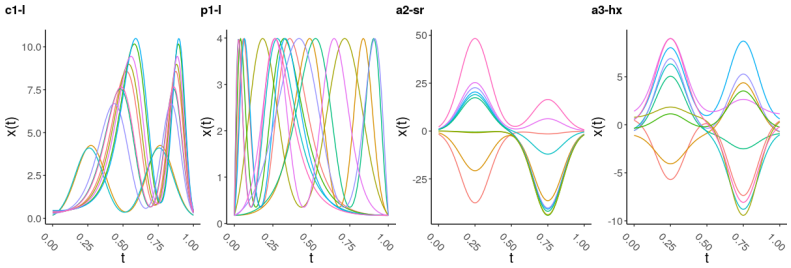

Figure 2: Example functions with 1 df coupled joint amplitude and phase variation (c1-l), 1 df phase variation (p1-l), and 2 and 3 df amplitude variation (a2-sr and a3-hx). c1-l and p1-l are based on linear, a2-sr and a3-hx are based on nonlinear parameter manifolds.

If no warping parameter is selected, identity warping is applied. Varying both amplitude *and* warping parameters in a setting induces joint amplitude and phase variation. Dependencies between amplitude and phase parameters induce dependencies between amplitude and phase variation.

The active parameters are either drawn uniformly from a linear parameter space, i.e., the manifold Θ is a linear subspace, or from a nonlinear parameter space, i.e, a (nonlinear) manifold. Note that the power function and the beta cdf warping are nonlinear transformations of the domain, thus we obtain a non-linear functional data manifold even if the parameter space Θ is linear. In the linear case we let $a_i, p_j \sim U[0.5, 3]$, $i = \{1, ..., 4\}$ and $j = \{2, 3, 4\}$, and $p_1 \sim U[0.01, 0.99]$. In the nonlinear case, parameter values are drawn uniformly from one of five different manifolds: the Swiss roll (1D and 2D), the 1D helix, the 2D S-curve, and the 2D two-peaked (tp) surface as provided by the R package `dimRed` [19]. For each setting, 1000 functional observations are generated based on 200 grid points. A summary of the simulation settings is provided in Table 2 and Figure 2 displays samples of functions with phase variation, amplitude variation, and joint, coupled amplitude and phase variation.

3.2 Results

The results show that functional data can – in principle, i.e. given the ground truth to compare against – be embedded with methods developed primarily for images or tabular data.

To start with we also sketch the effect of nonlinear warping in the next subsection. For that, we consider settings a1-l, p1-l, c1-l, a2-l, p2-l, i2-l with simple parameter spaces $\mathbb{R}^d$ first. Based on embeddings of the reference method MDS we analyze some specific pitfalls which result from the drawbacks of the performance measures described in Section 2.2. We finally turn to the settings with more complex parameter spaces, a2-sr, a3-hx, a3-sr, a3-sc, a3-tp, additionally evaluating tuning approaches based on the surrogate performance measures. We show that based on the correct tuning approach, it is possible to (automatically) obtain high-quality embeddings for these settings as well.

3.2.1 Embedding functional data with phase and amplitude variation

In this section, we highlight two essential aspects. First of all, the findings indicate that it is possible to successfully embed functional data using manifold methods, at least in these simple settings. In addition, we provide evidence that things can get rather complicated quickly if warping comes into play even if the underlying parameter space is a simple linear one. That said, the settings we consider here, a1-l, p1-l, c1-l, a2-l, p2-l, i2-l, include amplitude as well as phase variation and also both coupled and independent joint phase and amplitude variation.
As can be seen in Figure 3, ISOMAP is particularly successful. Clearly, perfect linear embeddings are achieved in settings with one degree of freedom and two degrees of freedom alike (a1-l, p1-l, c1-l, a2-l). Note that phase variation is induced by a nonlinear polynomial warping function in the setting with coupled phase and amplitude variation c1-l. Nevertheless, the functional manifold can be perfectly unfolded into its underlying linear structure. This is not the case for setting p2-l, where phase variation is induced by the Beta cdf. Here, the resulting structure of the functional manifold becomes more challenging to unfold into $\mathbb{R}^2$. This shows why embedding functional data can be especially complex and why it is important to use simple, low-dimensional parameter spaces for this study: the functional manifolds we are dealing with become nonlinear even though they are based on a deceptively simple parameter space. Moreover, consider setting i2-l, which has independent amplitude and phase variation (based on power warping). Even though the data are embedded nearly linearly, the distribution of the parameter values of the amplitude variation, indicated by colour code, no longer follows a simple linear direction in the embedding space. This also indicates the more complex structure of the functional manifold induced by the power warping.

For the embeddings of the other methods, similar findings can be reported, however overall they are not as successful as the ISOMAP embeddings. In the 1-dimensional settings, most embeddings are not perfectly linear. Moreover, unlike t-SNE, UMAP and DIFFMAP fail to recover setting i2-l respectively i2-l, p2-l, and a2-l. To sum up, manifold methods apparently seem to be able to recover the manifold structure of functional data, but how successfully they do so strongly depends on the complexity induced by the transformation of the parameter space. Structured phase variation can quickly lead to very complex functional manifolds which may not be embeddable faithfully in the sense that the functional data manifold is no longer isometric to a

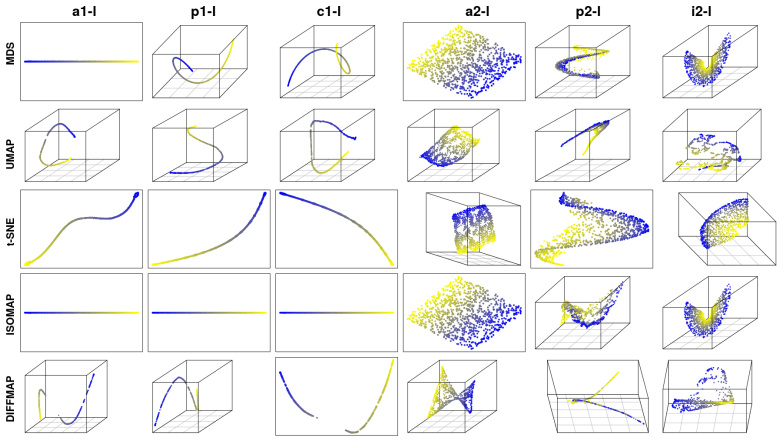

Figure 3: Embeddings for settings a1-l, p1-l, c1-l, a2-l, p2-l, i2-l. Color scale encodes the value of the first parameter in Θ. Since units for the embeddings are arbitrary, we omit axis labels here and in the following figures to save space.

low-dimensional linear subspace even if it is based on a very simple linear parameter space.

3.2.2 Pitfalls of surrogate performance measures

As outlined in section 2.2, the proposed surrogate performance measures suffer from some drawbacks. Here we show that one of the most worrisome resulting pitfalls is that these measures can frequently indicate high performance values even if the embedding is not of high quality in terms of ground-truth performance in the underlying parameter space.
To demonstrate the issue we concentrate on MDS embeddings of the nonlinear settings a2-sr, a3-hx, a3-sr, a3-sc, a3-tp. As can be seen in Figure 4, simple MDS is not able to *unfold* the functional manifolds into embeddings on linear subspaces or the circle. However, assessing the embedding, e.g. of setting a3-hx, using $AUC^{dir}_{R_{NX}}$ based on direct L_2-distance neighborhoods in the function space $\mathcal{F}$, i.e. the standard way of calculating distances, would indicate a perfect embedding quality of 1. However, assessing the embedding based on direct-distance-based neighborhoods in the parameter space Θ yields a much lower $AUC^{dir}_{R_{NX}}$ of only 0.78. Computing $AUC^{geo}_{R_{NX}}$, i.e. computing neighborhoods using geodesic distances, in the parameter space – recall that this is assumed to be the appropriate way to capture long-range distances on the manifold – leads to a further reduction, with an $AUC^{geo}_{R_{NX}}$ of only 0.553. This corresponds more closely to the visual impression since MDS is not able to unfold the intrinsic structure correctly.

So we see that naive application of standard performance measures can indicate high-quality embeddings even if the manifold is not accurately recovered at all – at least if *recovering* is defined as also *unfolding* non-linear manifolds. The example also shows

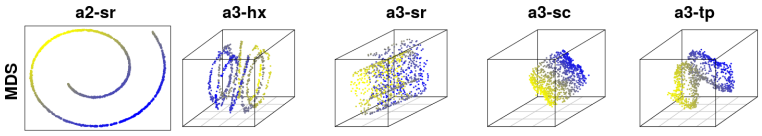

Figure 4: MDS embeddings for settings with nonlinear parameter space. Color scale encodes value of the first parameter in Θ.

that the assessment of the quality of an embedding provided by these performance measures is highly sensitive to the choice of the distance metric used to determine the neighborhood structure in the space against which the embedding space neighborhoods are compared.

These two pitfalls make the assessment of real data embeddings using surrogate performance measures particularly challenging since, in reality, the intrinsic structure is obviously unknown. In particular, automatic tuning approaches based on these measures have to be chosen and evaluated very carefully.

3.2.3 Evaluation of automatic parameter tuning

To assess the overall approach of tuning and embedding functional data, we concentrate on the more challenging nonlinear settings a2-sr, a3-hx, a3-sr, a3-sc, a3-tp. Since we want to assess the ability to detect and unfold intrinsic structure induced by a nonlinear parameter space here, identity warping is applied in all settings. To thoroughly evaluate the effects of the distance metric, we tuned each method on each setting based on $AUC^{dir}_{R_{NX}}$ and $AUC^{geo}_{R_{NX}}$, both based on the function space as well as the ground truth parameter space. That is, in total each method has been tuned four times for each setting. Computing performance based on the parameter and the function space allows us to compare what is *theoretically* possible – based on the ground truth parameter space – on the one hand, and what is *practically* feasible – based on the observable function space – on the other hand. Reliable tuning and evaluation methods based on the functional data should provide similar answers and results as those achieved by tuning and evaluating on the true underlying parameter space.

Figures 5 and 6 display the resulting embeddings for the considered settings. The embeddings have been obtained via tuning based on the parameter space, i.e., maximizing agreement between parameter space neighborhoods and embedding space neighborhoods in Figure 5 and obtained via tuning on the function space, i.e., maximizing agreement between function space neighborhoods and embedding space neighborhoods, in Figure 6. Successful embeddings should unfold these data either to a circle (a3-hx, 2nd column) or to linear subspaces. However, regarding the parameter space-optimized embeddings in Fig. 5, it becomes obvious that – even if the true parameter space is used – tuning can lead to embeddings that do not withstand visual inspection in that sense (e.g., see t-SNE and ISOMAP embeddings of a2-sr and a3-hx based on $AUC^{dir}_{R_{NX}}$; Fig. 5 A, second and third row). For setting a2-sr, $AUC^{dir}_{R_{NX}}$ indicates perfect embedding for ISOMAP and good embedding for t-SNE. The corresponding embeddings based on $AUC^{geo}_{R_{NX}}$ (Fig. 5 B, second and third row), however, withstand visual inspection far better. This already indicates that tuning based on geodesic distances can lead to better results than simply relying on direct distances. Turning to the function space

optimized embeddings (Fig. 6), we see that things can get even more involved in reality. Consider, for example, the ISOMAP embeddings again. The embeddings based on $AUC^{dir}_{R_{NX}}$ (Fig. 6 A) for a3-sc as well as for settings a2-sr, a3-hx, and a3-sr are not satisfactory, even though high performances are indicated by the measure.

These discrepancies between measured and actual performance are due to the fact that, in the case of direct distances, the performance measure is based on a suboptimal distance metric in the high-dimensional spaces Θ and $\mathcal{F}$. For strongly nonlinear settings, direct distances seem to be insufficient for tuning methods so that they correctly reflect the intrinsic structure. Analogously, this applies to the function space as well, since the L_2 metric in the function space is structurally very similar to Euclidean distance in a Euclidean space. So L_2-based neighborhoods are likely to be different from neighborhoods based on *geodesic* distances in the function space whenever the functional manifold is nonlinear. Due to the more complex structure of the function space, the effect seems to be intensified (for example, see ISOMAP embeddings for a3-sc: based on the parameter space, direct distances were sufficient to recover the manifold, whereas in the function space the intrinsic structure was only recovered if tuned based geodesic distances). Next to the effects of nonlinear domain warping, this is another example of the specific challenges of nonlinear dimension reduction in FDA settings.

Turning to the remaining methods, the picture is a little more difficult to make sense of, because the embeddings do not yield such clear differences as ISOMAP and t-SNE. In general, DIFFMAP and UMAP show arguably better embedding results based on $AUC^{geo}_{R_{NX}}$(e.g. a2-sr, a3-hx, Fig. 5), but, on the other hand, they benefit less from using geodesic distances for tuning (e.g. a3-sc, a3-hx, a2-sr, Fig. 6). DIFFMAP embeddings, in particular, differ the least among $AUC^{dir}_{R_{NX}}$ and $AUC^{geo}_{R_{NX}}$. Moreover, in some cases, the underlying manifold is hardly recognizable or not successfully unfolded, in particular, this holds for the DIFFMAP embeddings in settings a3-sr, a3-sc, and a3-tp.

To quantify the differences between using geodesic rather than direct distances to optimize and compute performance measures, Figure 7 shows the different optimal $AUC^{m}_{R_{NX}}$-values for all nonlinear settings obtained on the function space and the parameter space. The best values achieved based on the function space differ strongly from the ones based on the parameter space in several cases. In general, optimization via $AUC^{geo}_{R_{NX}}$ leads to smaller differences between tuning on function and parameter space distances. This is also reflected in Table 3, which shows the absolute differences $\Delta^m_{\bar{a}} := |\bar{a}^m_{ps} - \bar{a}^m_{fs}|$, with $\bar{a}^m_{ps}$ and $\bar{a}^m_{fs}$ the mean optimal $AUC^{m}_{R_{NX}}$ based on parameter respectively function space. The mean values $\bar{a}^m_{ps}$ and $\bar{a}^m_{fs}$ are computed over the ISOMAP, DIFFMAP, t-SNE, and UMAP embeddings and the settings a2-sr, a3-hx, a3-sc, a3-tp. Since setting a3-sr could not be embedded successfully with any of the methods even if optimized over the parameter space, it is excluded. Clearly, optimal $AUC^{geo}_{R_{NX}}$ (values based on the geodesic distances) differ less between function space and parameter space than $AUC^{dir}_{R_{NX}}$ (values based on the direct distances) for ISOMAP and t-SNE, while there is not much of a difference for UMAP and DIFFMAP. This is in line with the visual impression that ISOMAP and t-SNE yield clearly better embeddings based on tuning via $AUC^{geo}_{R_{NX}}$ for these settings.
This also indicates that it is frequently more appropriate to use geodesic distances – especially in function spaces – for performance assessment and tuning.

To sum up, we have seen that it is possible to obtain successful embeddings for functional data and that automatic parameter tuning can be applied in these simulation settings.

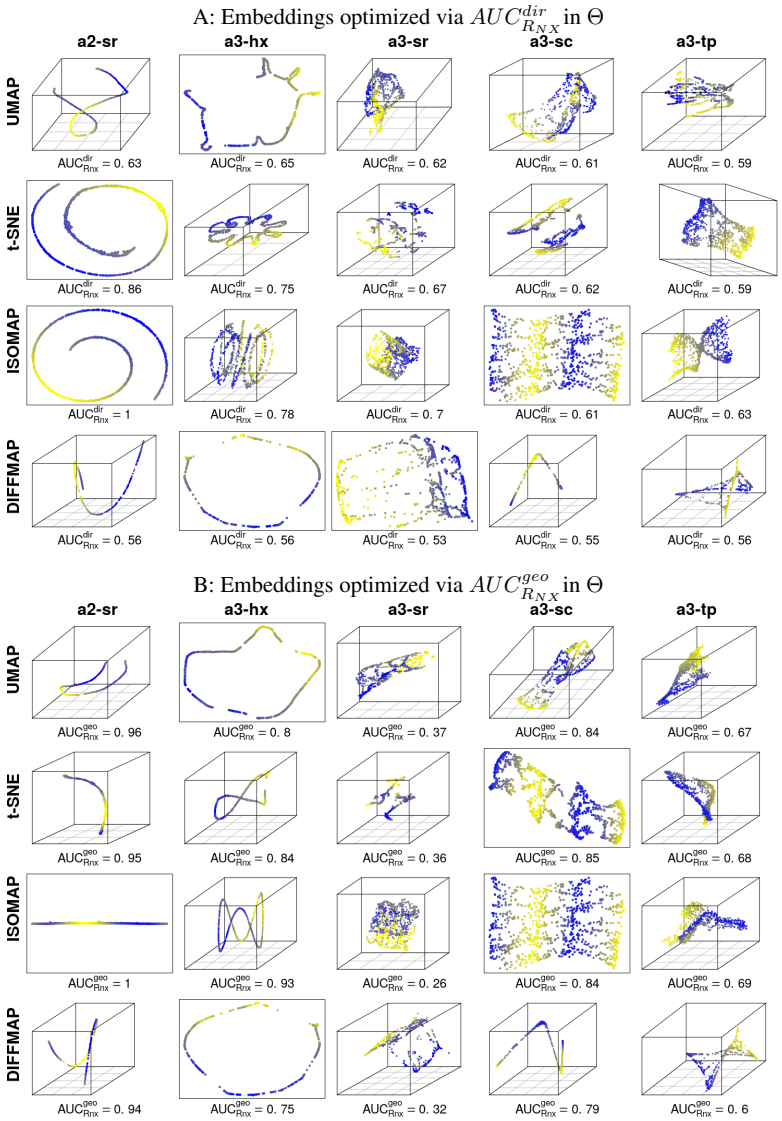

Figure 5: Parameter-space optimal embeddings of nonlinear settings a2-sr, a3-hx, a3-sr, a3-sc, a3-tp. A: first four rows based on parameter space $AUC^{dir}_{R_{NX}}$-optimization. B: lower four rows based on $AUC^{geo}_{R_{NX}}$. Color scale encodes the value of the first parameter in Θ.

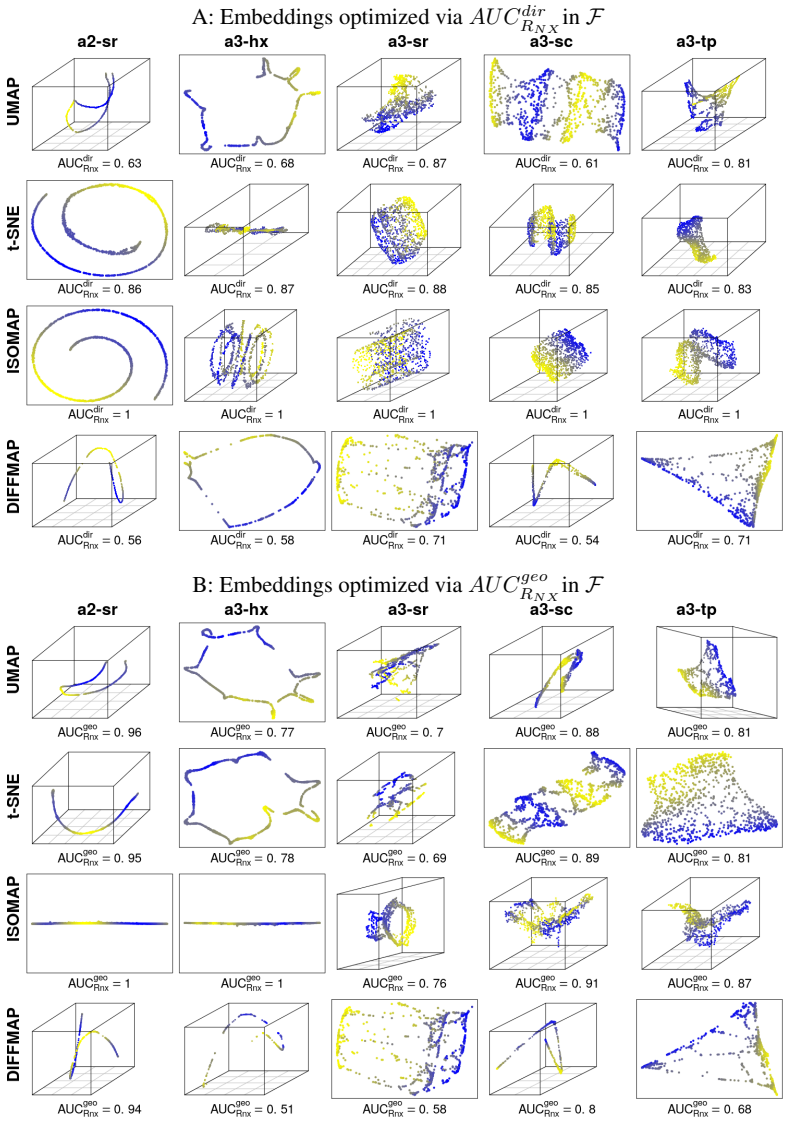

Figure 6: Functions-space optimal embeddings of nonlinear settings a2-sr, a3-hx, a3-sr, a3-sc, a3-tp. A: first four rows based on function space $AUC^{dir}_{R_{NX}}$-optimization. B: lower four rows based on $AUC^{geo}_{R_{NX}}$. Color scale encodes the value of the first parameter in Θ.

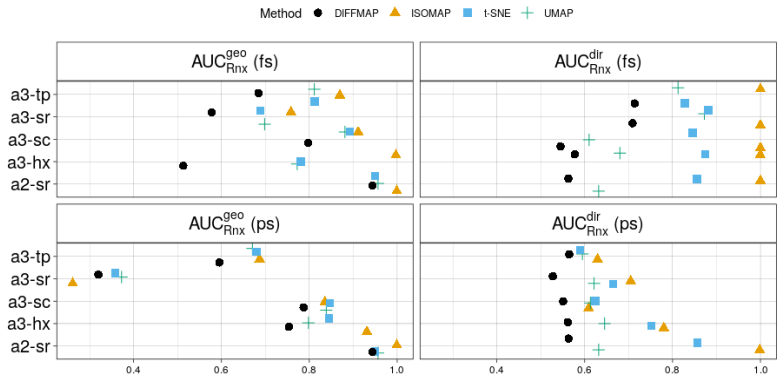

Figure 7: Function (fs) and parameter (ps) space optimal $AUC^m_{R_{NX}}$ values based on geodesic and direct distances for settings a2-sr, a3-hx, a3-sr, a3-sc, a3-tp.

Table 3: Comparing performance assessment based on geodesic and direct distances using absolute difference of mean $AUC^m_{R_{NX}}$ in parameter space and function space. Setting a3-sr is excluded because no visually appropriate embeddings could be obtained.

Method	Δ_a^{dir}	Δ_a^{geo}
ISOMAP	0.246	0.081
t-SNE	0.146	0.060
UMAP	0.064	0.052
DIFFMAP	0.085	0.043

Moreover, using distances in function space seems to be a reliable alternative to using distances in ground truth parameter space. In some of the complex settings a2-sr, a3-hx, a3-sr, a3-sc, a3-tp, however, tuning is successful only if based on geodesic distances rather than direct distances to define the neighborhoods in the high-dimensional space. In general, these are promising results indicating that (automatically) obtaining high-quality embeddings for real functional data is feasible. Yet, the approaches should not be applied lightly to real data. As we have seen, some of the methods may yield suboptimal or nonconforming embeddings even if tuned properly. Moreover, not every setting seems to be amenable to a successful embedding (e.g. see setting a3-sr) – factors that can lead to multiple nonconforming or misleading embeddings and overoptimistic or invalid conclusions if not assessed carefully.

4 Real data application

We now turn to real data examples in this section in order to verify the practical utility of the insights from our simulation study. As motivated in Section 2.4, we first apply our approach to two settings where the intrinsic structure of the data can be inferred from domain knowledge to a certain extent. Subsequently, we investigate a fully unsupervised real data example with completely unknown structure. In addition to $AUC^m_{R_{NX}}$, we also

evaluate embedding performance via Q_{local}^{m} here in order to investigate the distinctions between local and global performance measures. To begin with, we give a short description of all three date sets.

4.1 Data sets

We apply the embedding methods to two functional and one image data set. For two data sets, the COIL data and the earthquake data, the intrinsic structure can be inferred from prior knowledge in advance, at least to a certain extent. The intrinsic structure of the third data set, a spectrography data set, is not known. Figure 8 shows example observations of the three real data sets. More details are given in the sections devoted to the specific data set.

4.1.1 COIL data

COIL20 [30] is an image data set consisting of 128 x 128 pixel images of 20 objects. We use this data set albeit it is not functional, because it is a real data set for which the intrinsic structure can be inferred from substantial considerations and is nonlinear. For each object, 72 pictures were obtained by rotating the object around itself and taking a picture every 5 degrees of rotation. The end position equals the starting position and each picture reflects the object at a different angle. Thus, for each object, the COIL20 data set contains 72 observations with 16384 features containing single pixel intensities.

For this study, we use a subset of the COIL20 data containing only the pictures of the first object as the high-dimensional data to be embedded. This means the data set we use contains 72 pictures of the same object depicted in Figure 8 at different angles ranging from 0 to 360 degrees. Considering this setup, the 5-degree-picture should – for example – have approximately the same distance to the 0-degree-picture as the 355-degree-picture. The intrinsic structure of the data set is thus expected to be circular and one-dimensional, i.e., a setting supposed to be comparable to setting a3-hx.

Aside from these insights – which can be inferred from the original description of the data generating design and applies to all of the 20 COIL objects – for the specific object regarded here further considerations can be made, if one closely examines Figure 8. Due to the axial symmetry of the object, it appears to be more similar to itself at positions 0 and 180 degrees than at 90 and 270 degrees. This can be an indication of further existing structure which might be present in this specific example, but whether and how this is reflected in the embeddings is difficult to assess.

4.1.2 Earthquake data

The second real data set contains functional data of a seismological *in silico* experiment with both phase and amplitude variation described in [14]. It contains 1558 observations observed on 61 grid points. Each observation represents 60 seconds of absolute ground movement velocities at a virtual seismometer location for a simulated earthquake. The original investigation based on multivariate functional PCA of phase and amplitude variation revealed a two-dimensional linear structure of the data [14] reflecting the spatial distances of the virtual seismometers to each other and to the simulated earthquake's hypocenter. That is, from the analyses of the previous study we can infer that this is a data set with phase and amplitude variation and – following our framework – supposedly

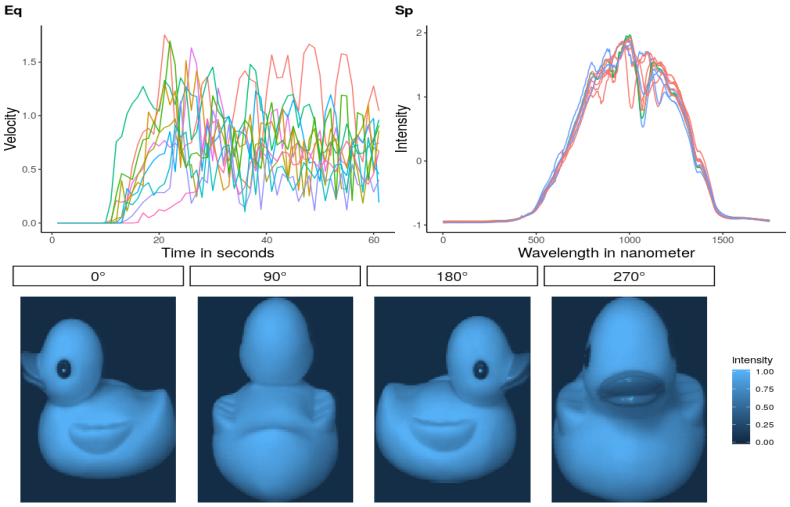

Figure 8: Upper row: Ten example observations of each the earthquake data (Eq) and the spectrography data (Sp). Lower row: Four images of the COIL object.

a relatively simple (linear) underlying parameter space, so a setting assumed to be comparable to settings i2-l or a2-l, for example.

4.1.3 Spectrography data

Finally, we consider another functional data set with 1004 functional observations observed on a grid of length 1751. The data was originally generated to investigate how forged spirits can be detected noninvasively via vibrational spectroscopy of the ethanol level, i.e. each observation is a spectrograph based on 1751 different wavelengths (see [21] for more details on the data set). The data is usually used as a classification problem and can be obtained[2] separated into a training set with 504 observations and a test set with 500 observations. Since we are in an unsupervised setting, we merged the training and test set and use the joined data set as the high-dimensional data to be embedded. Here, we cannot make any justifiable assumptions about the intrinsic structure and it is unclear what a successful embedding should look like.

4.2 Application to real data with known structure

Figure 9 displays the embeddings for the earthquake data and Figure 10 for the COIL data. The first two columns for each data set are obtained by tuning via the local performance measure Q^m_{local}, while the latter two columns are obtained by tuning via the global performance measure $AUC^m_{R_{NX}}$.

[2] `http://www.timeseriesclassification.com/description.php?Dataset=EthanolLevel`

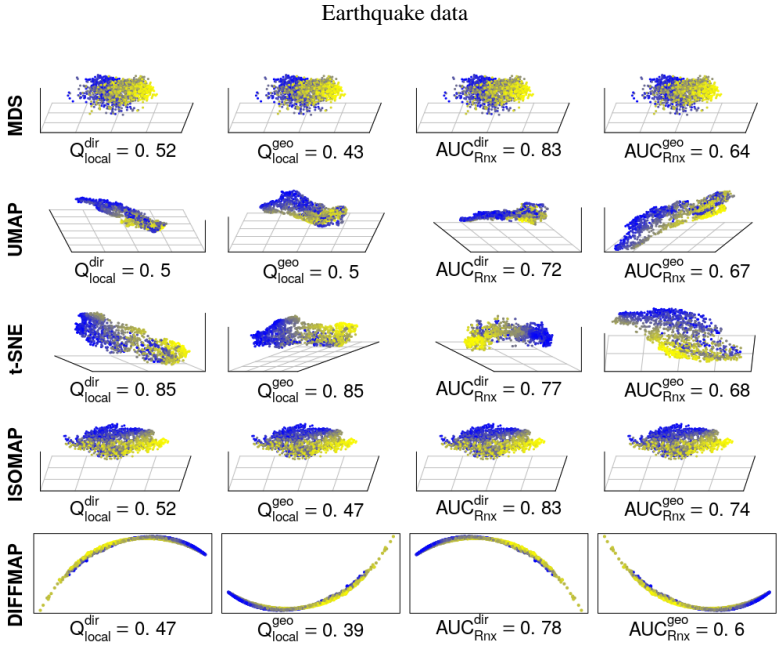

Figure 9: Embeddings of the earthquake data. First two columns obtained by optimization via Q^{m}_{local}, latter two columns via $AUC^{m}_{R_{NX}}$. Color scale encodes distance to hypocenter.

Most importantly, the results show that what has been observed for the simulated data also holds for real data: the low dimensional intrinsic structure – both linear and nonlinear – can be successfully embedded. Several embeddings of the COIL data show a 1-dimensional, circular structure, while a 2-dimensional linear structure results for the earthquake data. Moreover, closer examination reveals further interesting insights.

First of all, considering the earthquake data it appears that there is not much of a difference between embeddings based on $AUC^{geo}_{R_{NX}}$ and $AUC^{dir}_{R_{NX}}$ in a setting with a nearly linear intrinsic structure. The t-SNE and UMAP embeddings based on $AUC^{dir}_{R_{NX}}$ visually appear to be inferior to the ones based on $AUC^{geo}_{R_{NX}}$. Based on the performance values all embeddings based on $AUC^{geo}_{R_{NX}}$ perform somewhat worse. But in general, the performance differences are negligible. This is a desirable result in line with theoretical considerations: On approximately linear subspaces, geodesic distances ought to resemble direct distances.

Next to these findings, the COIL data opens up a rich pool of interesting insights that extend the understanding of embedding methods beyond that obtained in the simulation study. First of all, as outlined in Section 4.1.1, a 1-dimensional circular structure can

be assumed from the data generating procedure and this structure is detected by four of the embeddings. However, due to the symmetries of the rotating object, additional structure could be assumed and this additional structure seems to be recovered in several embeddings as well – in the case of some of the methods only if the global performance measure $AUC^m_{R_{NX}}$ is used for tuning, however. Consider the t-SNE embeddings where the effect is most prominent. In the two rightmost columns tuned for $AUC^m_{R_{NX}}$, the structure is ellipsoid, while it is circular in the two leftmost columns tuned for $\text{Q}^m_{\text{local}}$. These nonconforming embeddings can be explained by the axial symmetry of the object. Due to the symmetry, the object is more equal to itself at positions 0 and 180 degrees than at 90 and 270 degrees, which is a *global* characteristic (locally, i.e. within a small range of rotation angles, the object looks similar to itself everywhere). The local performance measure $\text{Q}^m_{\text{local}}$ is not able to reflect this global characteristic of the data sufficiently and the aspect is lost in UMAP, t-SNE, and DIFFMAP embeddings if tuned based on $\text{Q}^m_{\text{local}}$. These examples demonstrate that global structural properties can easily be "lost in translation" if an embedding is not tuned properly. It also has to be emphasized that – in contrast to t-SNE – those UMAP embeddings which sufficiently preserve global structure do not simultaneously preserve local structure in this setting. The situation is a little different for ISOMAP. As can be seen in Figure 10, the global structure is recovered in the embeddings, both based on $\text{Q}^m_{\text{local}}$ and $AUC^m_{R_{NX}}$. However, an additional dimension is needed to reflect this in the embedding, since all embeddings result in a twisted ring with an upward bend at two positions opposed to each other. That is, ISOMAP is not able to fully recover the structure in this example in the lowest possible number of embedding dimensions (similiar to DIFFMAP). Considering the MDS embeddings of the COIL data, we see that they are almost completely equal to the ISOMAP embeddings (only the performance indicated by the values of $\text{Q}^m_{\text{local}}$ is slightly worse for MDS). A possible explanation is that due to the low amount of observations (only 72 for COIL), the geodesic distances do not differ sufficiently from the direct distances and ISOMAP basically reflects MDS embeddings. This is supported by a couple of insights that can be gained by contrasting these results with the results of the simulations study.

First of all, regarding the MDS embeddings of the COIL data and the a3-hx data, we see that in both situations the intrinsic structure (a twisted ring and the helix, respectively) is recovered in principle. Yet, while the intrinsic structure of a3-hx gets unfolded by ISOMAP, the same is not true for COIL. The most fundamental difference between these two examples is the number of observations (1000 for a3-hx, 72 for COIL), which points towards the conclusion that, based on the low number of observations, a sufficient shortest path graph cannot be constructed for the COIL data such that MDS would benefit from it compared to simply using direct distances. Moreover, additional experiments in which the number of observations was increased from 1000 to 5000 in setting a3-sr showed that increasing the number of observations can lead to embeddings that successfully unroll the manifold – the failure of the methods to do so in our original experiment in setting a3-sr is likely to be due to too few observations. Finally, the conclusion is also supported by the fact that there are in general almost no differences between the COIL embeddings tuned via $AUC^{dir}_{R_{NX}}$ and $AUC^{geo}_{R_{NX}}$ – neither visually nor quantitatively. Recall, however, that in the simulated settings there have been large differences and – despite the fact that t-SNE and UMAP are able to recover the global structure here – it was frequently crucial to use geodesic distances so that the intrinsic structure could be unfolded.

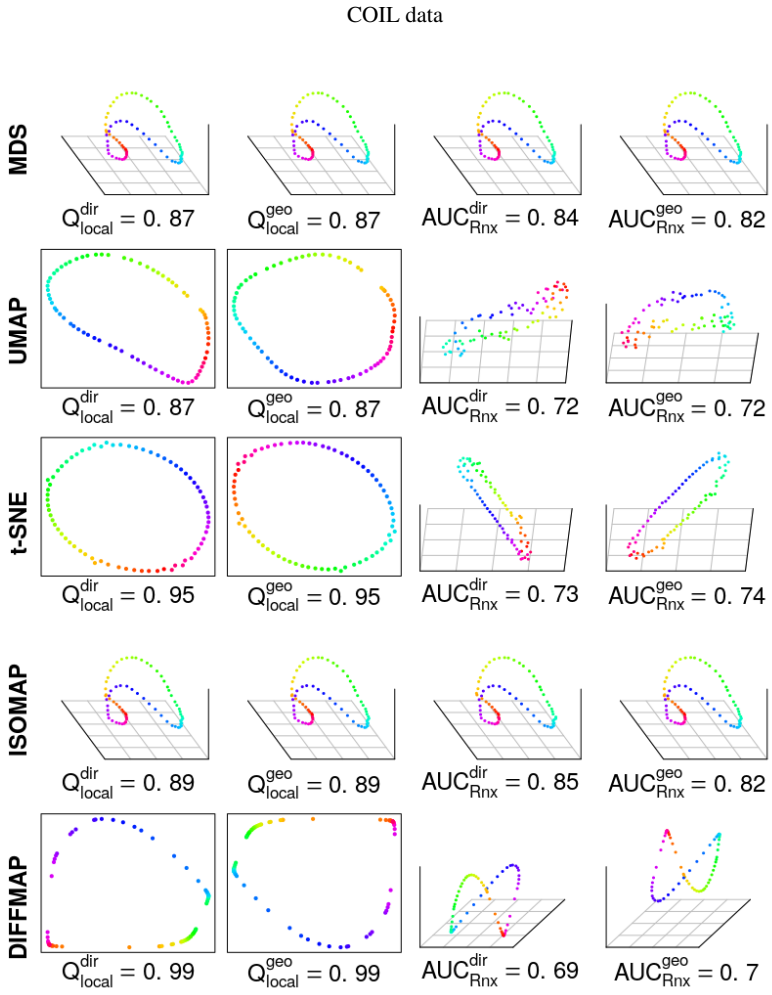

Figure 10: Embeddings of the COIL data. First two columns obtained by optimization via Q_{local}^{m}, latter two columns via $AUC_{R_{NX}}^{m}$. Color scale encodes rotation angle.

To sum up, the investigation of these two examples confirms that low dimensional intrinsic structure of real (functional) data can be automatically and successfully embedded. However, some pitfalls were clearly identified. For one, certain structural properties of the functional manifold can easily get lost in the embedding if it is not tuned with the correct strategy, which makes the assessment of fully unsupervised settings specifically challenging. Secondly, we saw that using geodesic instead of direct distances is only beneficial if the number of observations is sufficiently large in a nonlinear setting. Regardless, their use does not seem to be harmful even in settings with few observations and/or settings with linear structure of the functional manifold.

4.3 Application to real data with unknown structure

So far, we have seen that in several settings – simulated as well as real – where the intrinsic structure is known at least to a certain extent, functional data can be embedded successfully and that automatic tuning can be used to obtain suitably faithful embeddings. On the other hand, we identified some specific pitfalls. In this subsection, we illustrate the resulting challenges of nonconforming embeddings with a fully unsupervised real data example and point out an approach possibly allowing to gain some further insights in such fully unsupervised settings.

The embeddings of the spectrography data are depicted in Figure 11. The major problem is that there are – overall – two nonconforming structures that are detected. On the one hand, a closed, circular 3-dimensional structure – a “donut" –, detected by MDS, all ISOMAP embeddings except the one tuned via $AUC^{geo}_{R_{NX}}$, and arguably also the t-SNE embeddings based on $\mathrm{Q}^{m}_{\mathrm{local}}$. On the other hand, there is a curved, open 3-dimensional structure detected by ISOMAP based on $AUC^{geo}_{R_{NX}}$, t-SNE based on $AUC^{geo}_{R_{NX}}$ and to a lesser extent $AUC^{dir}_{R_{NX}}$, and UMAP. In this example, it is not possible to decide which of the embeddings better describes the true structure by visual inspection or reference to prior knowledge, nor is it expedient to simply maximize performance measures. E.g., $AUC^{m}_{R_{NX}}$ is similarly high for ISOMAP both based on geodesic as well as direct distances, yet they lead to nonconforming embeddings. The drawbacks and pitfalls of the performance measures described in the simulation study are fully apparent here. Although performance measures are available, deciding between nonconforming embeddings is far from straightforward.

However, the results gathered so far allow us to introduce additional decision criteria: First of all, we saw that, in the COIL and the a3-hx examples, closed structures were detected and recovered by all methods irrespective of the performance measures used for tuning. That is, in a setting with a not “fully” unfoldable structure that is closed in some way or the other, this structure was recovered in all cases considered. Specifically, that included embeddings obtained with $AUC^{geo}_{R_{NX}}$. On the other hand, if a nonlinear structure is ‘fully’ unfoldable, we saw that, in several cases, a fully unfolded embedding was achieved only if the embeddings were based on $AUC^{geo}_{R_{NX}}$ (e.g. see a3-sc) – provided there was enough data. Considering the spectrography data, we see that none of the embeddings based on $AUC^{geo}_{R_{NX}}$ – except MDS of course – show the closed circular structure. It may thus not be too far-fetched to infer that, if the intrinsic structure of this data set were closed and circular in reality, this would also be reflected by at least some of the embeddings based on $AUC^{geo}_{R_{NX}}$ (as was observed for example for the COIL and the a3-hx data) and that the “donut" does not actually resemble the true intrinsic structure sufficiently. In fact, one possible explanation could be that direct distances might falsely

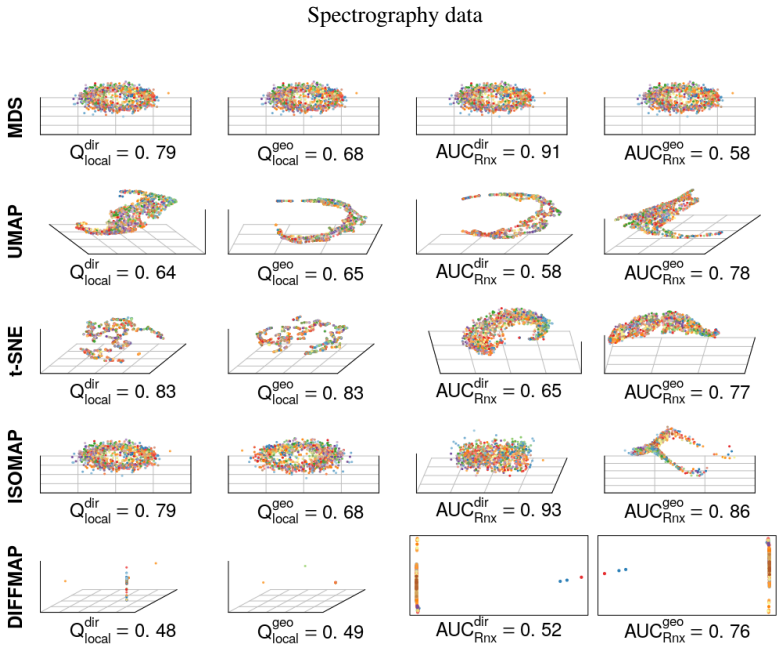

Figure 11: Embeddings of the spectrography data. First two columns obtained by optimization via Q^m_{local}, latter two via AUC^m_{RNX}. To improve visual differentiation, points are colored according to their observation number.

indicate that two objects at opposite ends of an open, but curved manifold are close together by taking a shortcut “through" the ambient space and thus connect these parts of the functional manifold in the embedding space. Similarly, based on local measures – that is, from a local perspective – such points might also appear close and global characteristics cannot be reflected sufficiently as was the case in the COIL example. This might result in unconnected manifold regions being pulled together, which might be appropriate locally but yields a wrong impression from a global perspective. The fact that UMAP, a method that is mainly concerned with an accurate representation of the local structure [29], does not show a closed structure even if based on Q^m_{local}, contributes to this conclusion. Note, however, that other explanations might be possible. For example, MDS embeddings reflected the true intrinsic structure in the simulated settings, which may also be the case here, and the other, “unclosed" structure might result from undiscovered effects of the approaches. Nevertheless, this example points in a direction to gaining insights into settings where so far no judgment is possible.

5 Discussion

Our results indicate that nonlinear dimension reduction methods can detect and "unfold" the manifold structure of functional data. For our settings, ISOMAP and t-SNE were seen to be particularly successful methods. Based on the same tuning regime, the other methods under comparison – DIFFMAP, UMAP, and MDS – frequently did not obtain similarly useful embeddings. However, we focused on recovery of the global data manifold structure in the embedding space. This is likely to affect UMAP adversely since it is optimized for high-fidelity reconstruction of local structures. The conclusion should thus not be that t-SNE or ISOMAP are superior for embedding functional data in general. Similar remarks apply for DIFFMAP, which is also a local method and performed less well than UMAP in our experiments. Especially in higher-dimensional real data settings, it frequently led to degenerate embeddings.

Furthermore, tuning strategies based on surrogate performance measures such as $AUC^m_{R_{NX}}$ should be applied with caution, since they may lead to very misleading embeddings with far from optimal configurations. In fact, we found evidence that embedding performance strongly depends on the distance metric m supplied to the performance measure used for tuning. Our results suggest that the use of geodesic distances – in particular in function space – is more likely to yield suitable embeddings if faithful representation of global structure is of importance: tuning embeddings based on the functional geodesic distances rather than direct distances yielded embeddings that were frequently much more similar to the ground truth structure in the simulation study and the expected structure in the real data examples.

Taking all insights obtained in this study into account, we propose to use the following nuanced approach to achieve more reliable embeddings of functional data: (1) Embeddings should be computed automatically by the described tuning approach. (2) At least two embedding methods should be included to account for different method performances. In addition, a tuning-free reference method should be included, for which we suggest MDS since it can recover intrinsic structures in simple cases although it does not "unfold" them. (3) Embeddings should be computed based on optimizing a local as well as a global performance measure. (4) Geodesic distances should be used. Tuning based on geodesic distances worked better for some methods, specifically in complex nonlinear settings, and did not have adverse effects on performance in any other settings. In addition, discrepancies between geodesic-based embeddings and direct-distance-based respectively local-performance-based embeddings can provide clues on the likely complexity of the intrinsic structure (closed vs. nonclosed, linear vs. nonlinear).

Moreover, some general questions are raised as well, as the last aspect strongly affects how to appropriately tackle specific unsupervised problems with manifold methods. For problems such as clustering or outlier detection, where preserving local structure can be sufficient, relying on non-geodesic distances can be appropriate. Yet, according to our findings, this can not simply be transferred to other tasks where global structure is more important, for example using manifold methods as a preprocessing or feature engineering step. In this setting, *unfolding* the manifold, i.e., detecting and simplifying the global structure, is important because reliable low dimensional representations not only improve visualization and exploration of functional data, but the embedding coordinates can also be exploited as features, which preserve the essential information contained in the functional data, for supervised learning (i.e., modeling and inference)

tasks. Using direct distances to assess embeddings quantitatively or to optimize learned embeddings in an automated fashion is (more) likely to lead to misleading results in this context.

In summary, our results show the potential of extending manifold methods for NDR to function-valued data, but also reveal challenges that are likely to come up in applications. In order to achieve reliable low-dimensional representations of functional data for visualization and exploration or to serve as feature inputs in supervised learning tasks, these issues will require additional attention by the research community. In future work, we will investigate the effect of noise-corrupted observations on the estimation of geodesic distances for functional data, since errors that shift observed functions off the functional manifold are likely to affect the recovery of geodesic distances adversely. In addition, the effects of grid resolution and data set size as well as the definition of alternative distance metrics that specifically account for certain characteristics of functions – for example, separate amplitude and phase distances [36] – are further important aspects.

More generally, this study should be considered in the light of a growing debate on replicability in methodological research. As has been outlined by many [4, 15, 17, 26, e.g.], methodological research claiming to show superior performance of its proposed methods and approaches in one way or another can frequently not be confirmed in independent replications. Our aim in this work was to provide a fairly neutral evaluation by design, pointing out specific pitfalls and drawbacks of several widely used manifold learning algorithms and possible meta-learning methods in a wide range of functional data settings. However, other evaluation frameworks are certainly possible and may yield additional insights and qualifications for our conclusions. As long as there is no general benchmarking and evaluation regime generally agreed upon, the results of all such studies will depend on the choice of this framework to some extent. In that regard, our study is also intended to serve as a starting point and we hope that it may contribute to initiating a discussion – similar to the ones in supervised learning and cluster analysis – on how to conduct neutral evaluations and foster replicable results in the important field of NDR and manifold learning.

Technical details

All code necessary to reproduce the experiment can be found on GitHub `https://github.com/HerrMo/fda-ndr`. To conduct the experiments we used R 3.6.3 [33] on a system with Linux Mint Cinnamon 19.2 and R packages `vegan` [31] for ISOMAP, `diffusionMap` [35] for DIFFMAP, `Rtsne` [20] for t-SNE, and `umap` [18] for UMAP. For the performance measures we used code of the functions `auc_rnx` and `q_local` from the `dimRed` package [19]. To compute MDS we used `cmdscale` of package `stats` and `isomapdist` of package `vegan` to compute geodesic distances.

Acknowledgments

This work has been funded 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.

Declarations of interest

None.

References

[1] Alaız, C.M.: Diffusion Maps Parameters Selection Based on Neighbourhood Preservation. Computational Intelligence (2015)

[2] Becht, E., McInnes, L., Healy, J., Dutertre, C.A., Kwok, I.W.H., Ng, L.G., Ginhoux, F., Newell, E.W.: Dimensionality reduction for visualizing single-cell data using UMAP. Nature Biotechnology **37**(1), 38–44 (2019)

[3] Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence **35**(8), 1798–1828 (2013)

[4] Boulesteix, A.L., Hoffmann, S., Charlton, A., Seibold, H.: A replication crisis in methodological research? Significance **17**(5), 18–21 (2020)

[5] Boulesteix, A.L., Lauer, S., Eugster, M.J.: A plea for neutral comparison studies in computational sciences. PloS one **8**(4), e61562 (2013)

[6] Cayton, L.: Algorithms for manifold learning. Tech. Rep. 1-17, Univ. of California at San Diego Tech. Rep (2005)

[7] Chen, D., Müller, H.G.: Nonlinear manifold representations for functional data. The Annals of Statistics **40**(1), 1–29 (2012)

[8] Chen, L., Buja, A.: Local Multidimensional Scaling for Nonlinear Dimension Reduction, Graph Drawing, and Proximity Analysis. Journal of the American Statistical Association **104**(485), 209–219 (2009)

[9] Coifman, R.R., Lafon, S.: Diffusion maps. Applied and Computational Harmonic Analysis **21**(1), 5–30 (2006)

[10] Dimeglio, C., Gallón, S., Loubes, J.M., Maza, E.: A robust algorithm for template curve estimation based on manifold embedding. Computational Statistics & Data Analysis **70**, 373 – 386 (2014)

[11] Eugster, M.J.: Benchmark Experiments: A Tool for Analyzing Statistical Learning Algorithms. Ph.D. thesis, LMU München (2011), `https://edoc.ub.uni-muenchen.de/12990/1/EugsterManuelJA.pdf`

[12] Goldberg, Y., Zakai, A., Kushnir, D., Ritov, Y.: Manifold learning: The price of normalization. Journal of Machine Learning Research **9**(Aug), 1909–1939 (2008)

[13] Gong, S., Boddeti, V.N., Jain, A.K.: On the Intrinsic Dimensionality of Image Representations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3987–3996 (2019)

[14] Happ, C., Scheipl, F., Gabriel, A.A., Greven, S.: A general framework for multivariate functional principal component analysis of amplitude and phase variation. Stat **8**(1), e220 (2019)

[15] Hutson, M.: Artificial intelligence faces reproducibility crisis. Science **359**(6377), 725–726 (2018)

[16] Kobak, D., Berens, P.: The art of using t-SNE for single-cell transcriptomics. Nature Communications **10**(1), 5416 (2019)

[17] Kobak, D., Linderman, G.C.: UMAP does not preserve global structure any better than t-SNE when using the same initialization. preprint, BioRxiv (2019), https://www.biorxiv.org/content/10.1101/2019.12.19.877522v1

[18] Konopka, T.: umap: Uniform Manifold Approximation and Projection (2020), https://CRAN.R-project.org/package=umap, R package version 0.2.4.1

[19] Kraemer, G., Reichstein, M., Mahecha, D., M.: dimRed and coRanking - Unifying Dimensionality Reduction in R. The R Journal **10**(1), 342 (2018)

[20] Krijthe, J.H.: Rtsne: T-Distributed Stochastic Neighbor Embedding using Barnes-Hut Implementation (2015), https://github.com/jkrijthe/Rtsne, R package version 0.15

[21] Large, J., Kemsley, E.K., Wellner, N., Goodall, I., Bagnall, A.: Detecting forged alcohol non-invasively through vibrational spectroscopy and machine learning. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining. pp. 298–309. Springer (2018)

[22] Lee, J.A., Renard, E., Bernard, G., Dupont, P., Verleysen, M.: Type 1 and 2 mixtures of Kullback–Leibler divergences as cost functions in dimensionality reduction based on similarity preservation. Neurocomputing **112**, 92–108 (2013)

[23] Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer Science & Business Media (2007)

[24] Lee, J.A., Verleysen, M.: Quality assessment of dimensionality reduction: Rank-based criteria. Neurocomputing **72**(7-9), 1431–1443 (2009)

[25] Liang, J., Chenouri, S., Small, C.G.: A new method for performance analysis in nonlinear dimensionality reduction. Statistical Analysis and Data Mining: The ASA Data Science Journal **13**(1), 98–108 (2020)

[26] Lucic, M., Kurach, K., Michalski, M., Gelly, S., Bousquet, O.: Are GANs created equal? A large-scale study. In: Advances in neural information processing systems. pp. 700–709 (2018)

[27] Ma, Y., Fu, Y.: Manifold learning theory and applications. CRC press (2011)

[28] Maaten, L.v.d., Hinton, G.: Visualizing Data using t-SNE. Journal of Machine Learning Research **9**, 2579–2605 (2008)

[29] McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. arXiv:1802.03426 (2018), https://arxiv.org/abs/1802.03426

[30] Nane, S., Nayar, S., Murase, H.: Columbia object image library: Coil-20. Dept. Comp. Sci., Columbia University, New York, Tech. Rep (1996)

[31] Oksanen, J., Blanchet, F.G., Friendly, M., Kindt, R., Legendre, P., McGlinn, D., Minchin, P.R., O'Hara, R.B., Simpson, G.L., Solymos, P., Stevens, M.H.H., Szoecs, E., Wagner, H.: vegan: Community Ecology Package (2019), https://CRAN.R-project.org/package=vegan, R package version 2.5-6

[32] Pfisterer, F., Beggel, L., Sun, X., Scheipl, F., Bischl, B.: Benchmarking time series classification – functional data vs machine learning approaches. arXiv:1911.07511 (2019), https://arxiv.org/abs/1911.07511

[33] R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2020), https://www.R-project.org/

[34] Ramsay, J.O., Silverman, B.W.: Functional data analysis. Springer series in statistics, Springer, New York, 2nd ed edn. (2005)

[35] Richards, J., Cannoodt, R.: diffusionMap: Diffusion Map (2019), `https://CRAN.R-project.org/package=diffusionMap`, R package version 1.2.0

[36] Srivastava, A., Wu, W., Kurtek, S., Klassen, E., Marron, J.S.: Registration of Functional Data Using Fisher-Rao Metric. arXiv:1103.3817 (2011), `http://arxiv.org/abs/1103.3817`

[37] Tenenbaum, J.B., Silva, V.d., Langford, J.C.: A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science **290**(5500), 2319–2323 (2000)

[38] Van Mechelen, I., Boulesteix, A.L., Dangl, R., Dean, N., Guyon, I., Hennig, C., Leisch, F., Steinley, D.: Benchmarking in cluster analysis: A white paper. arXiv:1809.10496 (2018), `https://arxiv.org/abs/1809.10496`

[39] Venna, J., Kaski, S.: Neighborhood Preservation in Nonlinear Projection Methods: An Experimental Study. In: Dorffner, G., Bischof, H., Hornik, K. (eds.) Artificial Neural Networks — ICANN 2001. pp. 485–491. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg (2001)

[40] Venna, J., Peltonen, J., Nybo, K., Aidos, H., Kaski, S.: Information retrieval perspective to nonlinear dimensionality reduction for data visualization. Journal of Machine Learning Research **11**(2) (2010)

[41] Wang, J.L., Chiou, J.M., Müller, H.G.: Functional data analysis. Annual Review of Statistics and Its Application **3**(1), 257–295 (2016)

[42] Wang, W., Yan, Y., Nie, F., Yan, S., Sebe, N.: Flexible manifold learning with optimal graph for image and video representation. IEEE Transactions on Image Processing **27**(6), 2664–2675 (2018)

8. A Geometric Perspective on Functional Outlier Detection

Chapter 8 deals with functional outlier detection. Based on a geometric perspective on the problem motivated by manifold learning, two different types of outliers – *off-manifold* and *on-manifold* – are differentiated. Extensive qualitative and quantitative experiments based on simulated and real-world data using MDS as a primary manifold learning method demonstrate the practical and theoretical utility of the approach. This includes comparisons with existing functional-data-specific methods.

Contributing article:

Herrmann, M., & Scheipl, F. (2021). A geometric perspective on functional outlier detection. Stats, 4(4), 971-1011. https://doi.org/10.3390/stats4040057

Copyright information:

Author contributions:

Moritz Herrmann had the idea of dealing with the topic in this way and wrote the paper. Fabian Scheipl made a substantial contribution by continuously revising the manuscript and adding ideas.

Supplementary material available at:

Code and data: https://github.com/HerrMo/fda-geo-out

stats

MDPI

Article

A Geometric Perspective on Functional Outlier Detection

Moritz Herrmann * and Fabian Scheipl

Department of Statistics, Ludwig-Maximilians-University, Ludwigstr. 33, 80539 Munich, Germany; fabian.scheipl@stat.uni-muenchen.de

* Correspondence: moritz.herrmann@stat.uni-muenchen.de; Tel.: +49-89-2180-3198

Abstract: We consider functional outlier detection from a geometric perspective, specifically: for functional datasets drawn from a functional manifold, which is defined by the data's modes of variation in shape, translation, and phase. Based on this manifold, we developed a conceptualization of functional outlier detection that is more widely applicable and realistic than previously proposed taxonomies. Our theoretical and experimental analyses demonstrated several important advantages of this perspective: it considerably improves theoretical understanding and allows describing and analyzing complex functional outlier scenarios consistently and in full generality, by differentiating between structurally anomalous outlier data that are off-manifold and distributionally outlying data that are on-manifold, but at its margins. This improves the practical feasibility of functional outlier detection: we show that simple manifold-learning methods can be used to reliably infer and visualize the geometric structure of functional datasets. We also show that standard outlier-detection methods requiring tabular data inputs can be applied to functional data very successfully by simply using their vector-valued representations learned from manifold learning methods as the input features. Our experiments on synthetic and real datasets demonstrated that this approach leads to outlier detection performances at least on par with existing functional-data-specific methods in a large variety of settings, without the highly specialized, complex methodology and narrow domain of application these methods often entail.

Keywords: functional data analysis; outlier detection; manifold learning; dimension reduction; multidimensional scaling; local outlier factors

Citation: Herrmann, M.; Scheipl, F. A Geometric Perspective on Functional Outlier Detection. *Stats* **2021**, *4*, 971–1011. https://doi.org/10.3390/stats4040057

Academic Editor: Manuel Oviedo de la Fuente

Received: 14 September 2021
Accepted: 12 November 2021
Published: 24 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

1. Introduction

1.1. Problem Setting and Proposal

Outlier detection for functional data is a challenging problem due to the complex and information-rich units of observations, which can be "outlying" or unusual in many different ways. Functional outliers are often categorized into magnitude and shape outliers [1,2], whereas Hubert et al. [3] differentiated between isolated and persistent outliers, the latter were further subdivided into shift, amplitude, and shape outliers. However, neither of these taxonomies yield precise, explicit, fully general definitions, which makes it difficult to theoretically describe, analyze, and compare functional outliers. Magnitude outliers, for example, have been defined as functional observations "outlying in some part or across the whole design domain" [1] (p. 1), or as "curves lying outside the range of the vast majority of the data" [2] (p. 2), whereas Hubert et al. [3] (p. 3) defined isolated outliers as observations that "exhibit outlying behavior during a very short time interval", in contrast to persistent outliers, which "are outlying on a large part of the domain".

To cut through the confusion, we propose a geometric perspective on functional outlier detection based on the well-known "manifold hypothesis" [4,5]. This refers to the assumption that ostensibly complex, high-dimensional data lie on a much simpler, lower-dimensional manifold embedded in the observation space and that this manifold's structure can be learned and then represented in a low-dimensional space, often simply called embedding space. We argue that such a perspective both clarifies and generalizes the

concept of functional outliers, without the need for any strong assumptions or prior knowledge about the underlying data-generating process or its outliers. In terms of theoretical development, the approach allows us to consistently formalize and systematically analyze functional outlier detection in full generality. We also demonstrate that procedures based on this perspective simplify and improve functional outlier detection in practice: this suggests a principled, yet flexible approach for applying well-established, highly performant standard outlier-detection methods such as local outlier factors (LOF) [6] to functional data, based on embedding coordinates obtained via manifold learning or dimension-reduction methods. Our experiments show that doing so performs at least on par with existing functional-data-specific outlier-detection methods, without the methodological complexity and limited applicability that methods specific to functional data often entail. Moreover, such lower-dimensional representations serve as an easily accessible visualization and exploration tool that helps uncover complex and subtle data structures that cannot be sufficiently reflected by one-dimensional outlier scores or labels, nor captured by many of the previously proposed 2D diagnostic visualizations for functional outliers.

1.2. Background and Related Work

Functional data analysis (FDA) [7] focuses on data where the units of observation are realizations of stochastic processes over compact domains. In many cases, the intrinsic dimensionality of functional data (FD) is much lower than the observed. First, while FD are infinite-dimensional in theory, they are high-dimensional in practice: functional observations are usually recorded on fine and dense grids of argument values. Second, the dominant drivers of the differences among functional observations are often comparatively low-dimensional, so that just a few modes of variation capture most of the structured variability in the data.

However, FD usually contain shape and translation, as well as phase variation, i.e., both "vertical" and "horizontal" variability. These different kinds of variability contribute to the difficulty of precisely defining and differentiating the various forms of functional outliers and developing methods that can "catch them all", making outlier detection a highly investigated research topic in FDA. For example, Arribas-Gil and Romo [2] argued that the proposed outlier taxonomy of Hubert et al. [3] can be made more precise in terms of expectation functions $f(t)$ and $g(t)$, with $f(t)$ a "common" process; see Figure 1.

$$
\begin{array}{l}
\textit{Functional outliers} \left\{ \begin{array}{l}
\textit{Isolated outliers}: \ g(t) = f(t) + \mathbb{1}\{t \in S\} h(t),\ S \subset U \\
\textit{Persistent outliers} \left\{ \begin{array}{l}
\textit{Magnitude outliers}: \ g(t) = \alpha f(t) \\
\textit{Shape outliers}: \ g \textit{ not related to } f \\
\textit{Shift outliers}: \ g(t) = \alpha + f(t),\ g(t) = f(t+\alpha),\ g(t) = f(h(t))
\end{array} \right.
\end{array} \right. \\
\\
\textit{Functional outliers} \left\{ \begin{array}{l}
\textit{Magnitude outliers} \left\{ \begin{array}{l} \textit{Isolated outliers} \\ \textit{Persistent outliers} \end{array} \right. \\
\textit{Shape outliers} \left\{ \begin{array}{l} \textit{Isolated outliers} \\ \textit{Persistent outliers} \end{array} \right.
\end{array} \right.
\end{array}
$$

Figure 1. Functional outlier taxonomies. Bottom: standard taxonomy. Top: the taxonomy as introduced by Hubert et al. Reprinted by permission from Springer Nature: Springer, Statistical Methods & Applications, Discussion of "Multivariate functional outlier detection", Arribas-Gil Ana, Romo Juan, Copyright 2015.

Despite these attempts, some fundamental issues remain unsolved. The proposed taxonomies do not provide precise definitions, and some of the definitions are contradictory to some extent. Finally, many outlier scenarios for realistic data-generating processes are not covered by the described taxonomies at all. As Arribas-Gil and Romo [2] themselves pointed out that settings with phase-varying data (i.e., "horizontal" variability through elastic deformations of the functions' domains) are not sufficiently reflected, as functions

deviating in terms of phase may be considered as shape outliers in cases where there are only a few such functions, but not in settings where all functions display such variation.

In addition, the taxonomy in Figure 1 provides a reasonable conceptual framework only if the nonoutlying data from the "common" data-generating process is characterized adequately just by its global mean function. This cannot be assumed for many real datasets, which often contain highly variable sets of functions, which display several modes of phase, shape, and/or amplitude variation simultaneously and/or come from multiple classes with class-specific means and higher moments (see Figure 5).

Published research focuses mostly on the development of outlier detection methods specifically for functional data, and a multitude of methods based on a variety of different concepts such as functional data depths [8,9], functional PCA [10], functional isolation forests [11], robust functional archetypoids [12], or functional outlier metrics such as directional outlyingness [13,14], often narrowly focused on detecting specific kinds of functional outliers, have been put forth. Dai et al. [1] proposed a transformation-based approach to functional outlier detection and claimed that sequentially transforming shape outliers, which "are much more challenging to handle", into magnitude outliers makes them easier to detect with established methods [1] (p. 2). The approach allows defining functional outliers more precisely in terms of the transformations being used, such as normalizing or centering functions or taking their derivatives, but practitioners still need to be able to come up with appropriate transformations for the data at hand first.

Recently, Xie et al. [15] introduced a decomposition of functional observations into amplitude, phase, and shift components, based on which specific types of outliers can be identified in a more general geometric framework without necessarily requiring functional data to be of comparatively low rank. Similar in spirit to our proposal, Hyndman and Shang [16] used kernel density estimation and half-space depth contours of two-dimensional robustified FPCA scores to construct functional boxplot equivalents and detect outliers, and Ali et al. [17] used data representations in two dimensions obtained from manifold methods for outlier detection and clustering, but the focus of both was on practicalities without considering the theoretical implications and general applicability of embedding-based approaches, nor did they consider the necessity of higher-dimensional representations. While Hyndman and Shang's HDR boxplots were based on a similar combination of methods as our approach, they did not consider their geometrical foundations and, thus, did not make use of their full potential, firstly by considering only the two largest PCs and secondly by dichotomizing observations into outliers and inliers instead of providing continuous scores of outlyingness. Yu et al. [18] developed a test statistic for outlier detection based on the observed maxima of scaled PC score vectors, i.e., outlyingness defined in terms of a single mode of variation. However, this NHST framework for outlier detection needs to assume both that the common data have a *single* consistent mean function and that all deviations from this mean function are i.i.d. realizations of a mean-zero Gaussian process. Both of these assumptions seem highly restrictive to us and are likely to be untenable in many real-world applications.

The remainder of the paper is structured as follows: We provide a theoretical formalization and discussion of our geometric approach in Section 2. Based on these theoretical considerations, Section 3 presents extensive experiments. Section 3.1 covers a detailed qualitative analysis of real-world data, while Section 3.2 provides quantitative experiments and systematic comparisons to previously proposed methods on complex synthetic outlier scenarios. We conclude with a discussion in Section 4.

2. Functional Outlier Detection as a Manifold-Learning Problem

In this section, we first define two forms of functional outliers from a geometric view point: off- and on-manifold outliers. We then illustrate how this perspective contains and extends existing outlier taxonomies and how it can be used to formalize a large variety of additional scenarios for functional data with outliers.

2.1. The Two Notions of Functional Outliers: Off- and On-Manifold

Our approach to functional outlier detection rests on the manifold assumption, i.e., the assumption that observed high-dimensional data are intrinsically low-dimensional. Specifically, we put forth that observed functional data $x(t) \in \mathcal{F}$, where $\mathcal{F}$ is a function space, arise as the result of a mapping $\phi : \Theta \to \mathcal{F}$ from a (low-dimensional) parameter space $\Theta \subset \mathbb{R}^{d_2}$ to $\mathcal{F}$, i.e., $x(t) = \phi(\theta)$. Conceptually, a d_2-dimensional parameter vector $\theta \in \Theta$ represents a specific combination of values for the modes of variation in the observed functional data, such as level or phase shifts, amplitude variability, class labels, and so on. These parameter vectors are drawn from a probability distribution P over $\mathbb{R}^{d_2}$: $\theta_i \sim P \ \forall \ \theta_i \in \Theta$, with $\Theta = \{\theta : f_P(\theta) > 0\}$ and f_P the density to P. Mapping this parameter space to the function space creates a functional manifold $\mathcal{M}_{\Theta,\phi}$ defined by ϕ and Θ: $\mathcal{M}_{\Theta,\phi} = \{x(t) : x(t) = \phi(\theta) \in \mathcal{F}, \theta \in \Theta\} \subset \mathcal{F}$, and an example is depicted in Figure 2. For $\mathcal{F} = L^2$ with data from a single functional manifold that is isomorphic to some Euclidean subspace, Chen and Müller [19] developed the notions of a manifold mean and modes of variation. Similarly, Dimelgio et al. [20] developed a robust algorithm for template curve estimation for connected smooth submanifolds of $\mathbb{R}^d$.

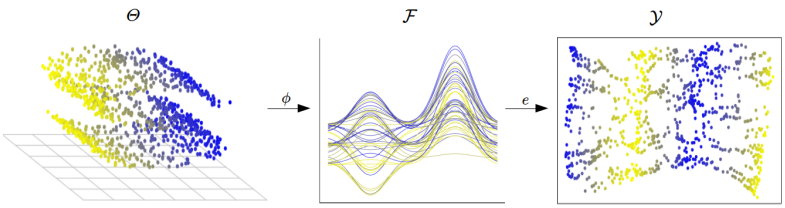

Figure 2. Functional data from a manifold-learning perspective. Image source: Herrmann and Scheipl [21]; use permitted under the Creative Commons Attribution License CC BY-SA 4.0.

Unlike these single-manifold settings, our conceptualization of outlier detection is based on two functional manifolds. That is, we assume a dataset $X = \{x_1(t), \ldots, x_n(t)\}$ with n functional observations coming from two separate functional manifolds $\mathcal{M}_c = \mathcal{M}_{\Theta_c,\phi_c}$ and $\mathcal{M}_a = \mathcal{M}_{\Theta_a,\phi_a}$, with $\mathcal{M}_j \subset \mathcal{F}$, $j \in \{c, a\}$ and $X \subset \{\mathcal{M}_c \cup \mathcal{M}_a\}$, with $\mathcal{M}_c$ representing the "common" data-generating process and $\mathcal{M}_a$ containing anomalous data. Moreover, for the purpose of outlier detection and in contrast to the settings with a single manifold described in the referenced literature, we are less concerned with precisely approximating the intrinsic geometry of each manifold. Instead, it is crucial to consider the manifolds $\mathcal{M}_c$ and $\mathcal{M}_a$ as submanifolds of $\mathcal{F}$, since we require not just a notion of distance between objects on a single manifold, but also a notion of distance between objects on different manifolds using the metric in $\mathcal{F}$. Note that function spaces such as $\mathcal{C}$ or L^2, which are commonly assumed in FDA [22], are naturally endowed with such a metric structure. Both $\mathcal{C}(D)$ and all $L^p(D)$ spaces over compact domain D are Banach spaces for $p \geq 1$ and, thus, also metric spaces [23].

Finally, we assume that we can learn from the data an embedding function $e : \mathcal{F} \to \mathcal{Y}$ that maps observed functions to a d_1-dimensional vector representation $y \in \mathcal{Y} \subset \mathbb{R}^{d_1}$ with $e(x(t)) = y$, which preserves at least the topological structure of $\mathcal{F}$, i.e., if $\mathcal{M}_c$ and $\mathcal{M}_a$ are unconnected components of $\mathcal{F}$, their images under e are also unconnected in $\mathcal{Y}$ and ideally yield a close approximation of the ambient geometry of $\mathcal{F}$.

Definition 1. *Off- and on-manifold outliers in functional data.*

Without loss of generality, let $r = \frac{|\{x_i(t):x_i(t)\in\mathcal{M}_a\}|}{|\{x_i(t):x_i(t)\in\mathcal{M}_c\}|} \lll 1$ be the outlier ratio, i.e., most observations are assumed to stem from $\mathcal{M}_c$. Furthermore, let Θ_c and Θ_a follow the distributions P_c and P_a, respectively. Let $\Omega^*_{\alpha,P}$ be an α-minimum volume set of P for some $\alpha \in (0, 1)$, where $\Omega^*_{\alpha,P}$ is defined as a set minimizing the quantile function

$V(\alpha) = \inf_{C \in \mathcal{C}}\{\text{Leb}(C) : P(C) \geq \alpha\}, 0 < \alpha < 1\}$ for i.i.d. random variables in $\mathbb{R}^d$ with distribution P, $\mathcal{C}$ a class of measurable subsets in $\mathbb{R}^d$, and Lebesgue measure Leb [24], i.e., $\Omega^*_{\alpha,P}$ is the smallest region containing a probability mass of at least α.

A functional observation $x_i(t) \in X$ is then:

- An off-manifold outlier if $x_i(t) \in \mathcal{M}_a$ and $x_i(t) \notin \mathcal{M}_c$;
- An on-manifold outlier if $x_i(t) \in \mathcal{M}_c$ and $\theta_i \notin \Omega^*_{\alpha,P_c}$.

To paraphrase, we assume that there is a single "common" process generating the bulk of observations on $\mathcal{M}_c$ and an "anomalous" process defining structurally different observations on $\mathcal{M}_a$. We follow the standard notion of outlier detection in this, which assumes that there are two data-generating processes [1,25,26]. Note that this does not necessarily imply that off-manifold outliers are similar to each other in any way: P_a could be very widely dispersed and/or $\mathcal{M}_a$ could consist of multiple unconnected components representing different kinds of anomalous data. The essential assumption here is that the process from which most of the observations are generated yields structurally relatively similar data. This is reflected by the notion of the two manifolds $\mathcal{M}_c$ and $\mathcal{M}_a$ and the ratio r. We consider settings with $r \in [0, 0.1]$ as suitable for outlier detection. By definition, the number of on-manifold outliers, i.e., distributional outliers on $\mathcal{M}_c$ as opposed to the structural outliers on $\mathcal{M}_a$, only depends on the α-level for Ω^*_{α,P_c}.

Note that outlyingness in functional data is often defined only in terms of shape or magnitude, but the concept ought to be conceived much more generally. The most important aspect from a practical perspective is that any kind of structural difference will be reliably reflected in low-dimensional representations that can be learned via manifold methods, as we show in Section 3. These methods yield embedding coordinates $y \in \mathcal{Y}$ that capture the structure of data and their outliers.

2.2. Methods

To illustrate some of the implications of our general perspective on functional outlier detection and showcase its practical utility, we mostly use metric multidimensional scaling (MDS) [27] for dimension reduction and local outlier factors (LOF) [6] for outlier scoring in the following. Note, however, that the proposed approach is not at all limited to these specific methods, and many other combinations of outlier detection methods applied to lower-dimensional embeddings from manifold-learning methods are possible. However, MDS and LOF have some important favorable properties: First of all, both methods are well understood and widely used and tend to work reliably without extensive tuning since they do not have many hyperparameters. Specifically, LOF only requires a single parameter `minPts`, which specifies the number of nearest neighbors used to define the local neighborhoods of the observations, and MDS only requires specification of the embedding dimension.

More importantly, our geometric approach rests on the assumption that functional outlier detection can be based on some notion of distance or dissimilarity between functional observations, i.e., that abnormal or outlying observations are separated from the bulk of the data in some ambient (function) space. As MDS optimizes for an embedding, which preserves all pairwise distances as closely as possible (i.e., tries to project the data isometrically), it also retains a notion of the distance between unconnected manifolds in the ambient space. This property of the embedding coordinates retaining the ambient space geometry as much as possible is crucial for outlier detection. This also suggests that manifold-learning methods such as ISOMAP [28], t-SNE [29], or UMAP [30], which do not optimize for the preservation of ambient space geometry via isometric embeddings by default, may require much more careful tuning in order to be used in this way. Our experiments support this theoretical consideration, as can be see in Figure 11. For LOF, this implies that larger values for `minPts` are to be preferred here, since such LOF scores take into account more of the global ambient space geometry of the data instead of only the local neighborhood structure. In Section 3, we show that `minPts` $= 0.75n$, with n the

number of functional observations in a dataset, seems to be a reliable and useful default for the range of datasets we consider.

Two additional aspects need to be pointed out here. First, throughout this paper, we compute most distances using the L_2 metric. This yields MDS coordinates that are equivalent to standard functional PCA scores (up to rotation). The proposed approach, however, is not restricted to L_2 distances. Combining MDS with distances other than L_2 yields embedding solutions that are no longer equivalent to PCA scores, and suitable alternative distance measures may yield better results in particular settings. We illustrate this aspect using the L_{10} metric and two phase-specific distance measures in Section 3.3, which we apply to simulated data with isolated outliers and a real dataset of outlines of neolithic arrowheads, respectively. Similarly, using alternative manifold-learning methods could be beneficial in specific settings, as long as they are able to represent not just the local neighborhood structure or on-manifold geometry, but also the global ambient space geometry.

Second, even though the LOF could also be applied directly to the dissimilarity matrix of a functional dataset without an intermediate embedding step, most anomaly-scoring methods cannot be applied directly to such distance matrices and require tabular data inputs. By using embeddings that accurately reflect the (outlier) structure of a functional dataset, any anomaly-scoring method requiring tabular data inputs can be applied to functional data as well. In this work, we apply LOF on MDS coordinates to evaluate whether functional data embeddings can faithfully retain the outlier structure. Furthermore, embedding the data before running outlier-detection methods often provides large additional value in terms of visualization and exploration, as the ECG data analysis in Section 3.1 shows.

2.3. Examples of Functional Outlier Scenarios

We can now give precise formalizations of different functional outlier scenarios and investigate the corresponding low-dimensional representations. In this section, we first show that the geometrical approach is able to describe existing taxonomies (see Figure 1) more consistently and precisely. We then illustrate its ability to formalize a much broader general class of outlier detection scenarios and discuss the choice of the distance metric and the dimensionality of the embedding.

2.3.1. Outlier Scenarios Based on Existing Taxonomies

Structure induced by shape: In the taxonomy depicted in Figure 1, top, the common data-generating process is defined by the expectation function $f(t)$. This can be formalized in our geometrical terms as follows: the set of functions defined by the "common process" $f(t)$ defines a functional manifold (in terms of shape), i.e., the structural component is represented by the expectation function of the common process. That means we can define $\mathcal{M}_c = \{x(t) : x(t) = \theta f(t) = \phi(\theta, t)\}$ or $\mathcal{M}_c = \{x(t) : x(t) = f(t) + \theta = \phi(\theta, t), \theta \in \mathbb{R}\}$. More generally, we can also model this jointly with $\mathcal{M}_c = \{x(t) : \theta_1 f(t) + \theta_2 = \phi(\theta, t), \theta = (\theta_1, \theta_2)' \in \mathbb{R}^2\}$. In each case, magnitude and (vertical) shift outliers as defined in the taxonomy correspond to on-manifold outliers in the geometrical approach, as such observations are elements of $\mathcal{M}_c$. Isolated and shape outliers, on the other hand, are by definition off-manifold outliers, as long as "g is not related to f" is specified as $g \neq \theta f \; \forall \, \theta \in \mathbb{R}$. For example, if we define $\mathcal{M}_a = \{x(t) : x(t) = \theta g(t)\}$, it follows that $\mathcal{M}_c \cap \mathcal{M}_a = \emptyset$. The same applies to isolated outliers, because $g(t) = f(t) + I_U(t)h(t) \neq \theta_1 f(t) + \theta_2$.

Figure 3 shows an example of such an outlier scenario taken from [8]. Following their notation, the two manifolds can be defined as $\mathcal{M}_c = \{x(t) | x(t) = b + 0.05t + \cos(20\pi t), b \in \mathbb{R}\}$ and $\mathcal{M}_a = \{x(t) | x(t) = a + 0.05t + \sin(\pi t^2), a \in \mathbb{R}\}$ with $t \in [0, 1]$ and $a \sim N(\mu = 5, \sigma = 4)$, $b \sim N(\mu = 5, \sigma = 3)$. Note that the off-manifold outliers lie within the mass of data in the visual representation of the curves, whereas in the low-dimensional embedding, they are clearly separable.

However, we argue that the way shape outliers are defined in Figure 1 is too restrictive, as many isolated outliers clearly differ in shape from the main data, but are not captured by the given definition if the shape is considered in terms of "g not related to f". In contrast, the geometrical perspective with its concepts of off- and on-manifold outliers reflects that consistently. Another issue with the considered taxonomy concerns horizontal shift outliers $f(t+\alpha)$ or $f(h(t))$. Aribas-Gil and Romo [2] specifically tackled that aspect in their discussion. They distinguished between situations where "all the curves present horizontal variation" (Case I), which is the no-outlier scenario for them, and situations where only a few phase-varying observations are present (Case II), which constitutes an outlier scenario. Again, the geometric perspective allows reflecting that consistently. In Appendix A, we make these two notions explicit by defining manifolds accordingly.

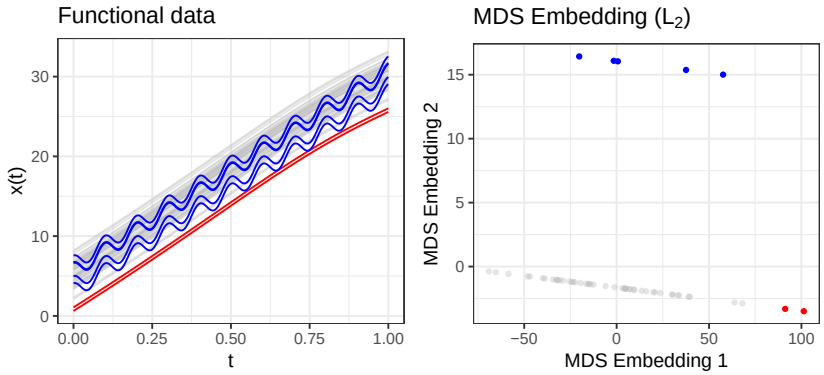

Figure 3. Functional outlier scenario ($n = 54, r = 0.09$) with shape variation inducing structural differences. Off-manifold outliers colored in blue; two on-manifold outliers colored in red.

2.3.2. General Functional Outlier Scenarios

As already noted, the concept of structural difference we propose is much more general. It is straightforward to conceptualize other outlier scenarios with an induced structure beyond shape. Consider the following theoretical example: take a parameter manifold $\Theta \subset [0,\infty] \times [0,\infty] \times [0,\infty] \times [0,\infty]$ and an induced functional manifold $\mathcal{M} = \{f(t); t \in [0,1] : f(t) = \theta_1 + \theta_2 t^{\theta_3} + I(t \in [\theta_4 \pm 0.1])\}$. Each dimension of the parameter space controls a different characteristic of the functional manifold: θ_1 the level, θ_2 the magnitude, θ_3 the shape, and θ_4 the presence of an isolated peak around $t = \theta_4$. One can now define a "common" data-generating process, i.e., a manifold $\mathcal{M}_c$, by holding some of the dimensions of Θ fixed and only varying the rest, either independently or not. On the other hand, one can define an "anomalous" data-generating process, i.e., a structurally different manifold $\mathcal{M}_a$, by letting those fixed in $\mathcal{M}_c$ vary, or simply setting them to values unequal to those used for $\mathcal{M}_c$, or by using different dependencies between parameters than for $\mathcal{M}_c$, e.g., if $\theta_1 = \theta_2$ for $\mathcal{M}_c$, let $\theta_1 = -\theta_2$ for $\mathcal{M}_a$. This implies that one can define data-generating processes so that any functional characteristic (level, magnitude, shape, "peaks", and their combinations) can be on-manifold or off-manifold outliers, depending on how the "common" data manifold $\mathcal{M}_c$ is defined.

Figure 4 shows a setting in which $\mathcal{M}_c$ is defined purely in terms of complex shape variation, while $\mathcal{M}_a$ contains vertically shifted versions of elements in $\mathcal{M}_c$: Let $\mathcal{M}_c$ be the functional manifold of Beta densities $f_B(t;\theta_1,\theta_2)$ with shape parameters $\theta_1, \theta_2 \in [1,2]$, and let $\mathcal{M}_a$ be the functional manifold of Beta densities with shape parameters $\theta_1, \theta_2 \in [1,2]$ shifted vertically by some scalar quantity $\theta_3 \in [0, 0.5]$, that is $\mathcal{M}_c = \{f(t); t \in [0,1] : f(t) = f_B(t;\theta_1,\theta_2)\}$ with $\Theta_c = [1,2]^2$ and $\mathcal{M}_a = \{f(t); t \in [0,1] : f(t) = f_B(t;\theta_1,\theta_2) + \theta_3\}$ with $\Theta_a = \Theta_c \times [0, 0.5]$.

As can be seen in Figure 4, both manifolds contain substantial shape variation that is identically structured, but those from $\mathcal{M}_a$ are also shifted upwards by small amounts. Note

that many shifted observations lie within the main bulk of the data on large parts of the domain. In the 2D embeddings based on unnormalized L_1-Wasserstein distances [31] (also know as the "Earth mover's distance", top right) and 3D embeddings based on standard L_2 distances (bottom right), we see that this structure is captured with high accuracy, even though it is hardly visible in the functional data, with most anomalous observations clearly separated from the common manifold data, whose embeddings are concentrated on a narrow subregion of the embedding space. An observation on $\mathcal{M}_a$ that is very close to $\mathcal{M}_c$, lying well within the main bulk of functional observations, also appears very close to $\mathcal{M}_c$ in both embeddings. This example shows that the two functional manifolds do not need to be completely disjoint, nor yield visually distinct observations for our approach to yield useful results. It also shows that the choice of an appropriate dissimilarity metric for the data can make a difference: a 2D embedding is sufficient for the more suitable Wasserstein distance, which is designed for (unnormalized) densities (top right panel), while a 3D embedding is necessary for representing the relevant aspects of the data geometry if the embedding is based on the standard L_2 metric (lower right panels). For a comparison with currently available outlier visualization methods for this example, see Figure A4 in Appendix D.

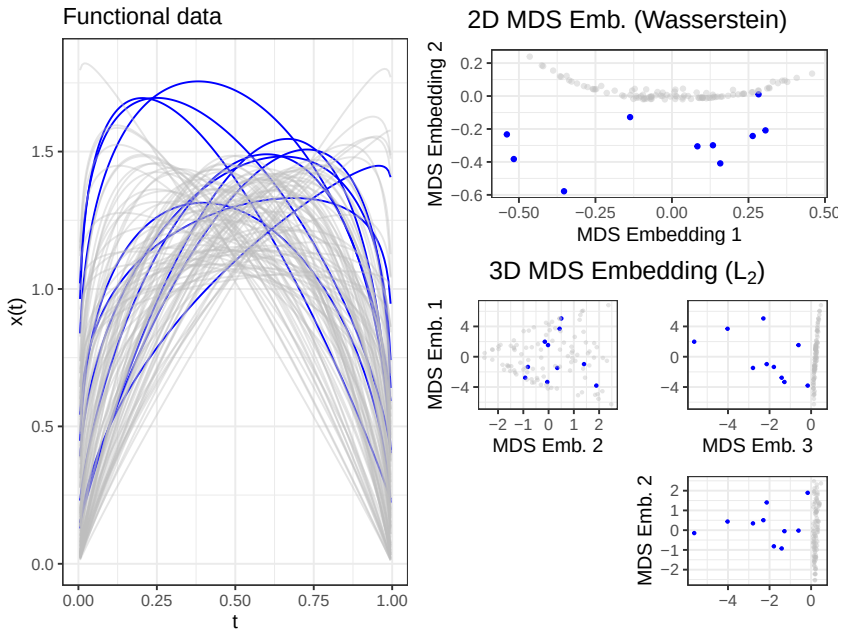

Figure 4. Functional outlier scenario ($n = 100, r = 0.1$) with vertical shifts inducing structural differences. MDS embeddings based on unnormalized L_1-Wasserstein distances and L_2 (Euclidean) distances on the right.

In summary, we propose that the manifold perspective allows defining and representing a very broad range of functional outlier scenarios and data-generating processes. We argue that these properties make the geometrical approach very compelling for functional data, because it is flexible, conceptualizes outliers on a much more general level (for example, structural differences not in terms of shape) than before, and allows theoretically assessing a given setting.

Beyond its theoretical utility of providing a general notion of functional outliers, it has crucial practical implications: outlier characteristics of functional data, in particular structural differences, can be represented and analyzed using low-dimensional representa-

Stats **2021**, *4* 979

tions provided by manifold-learning methods, regardless of which functional properties define the "common" data manifold and which properties are expressed in structurally different observations. From a practical perspective, on-manifold outliers will appear "connected", whereas off-manifold outliers will appear "separated" in the embedding, and the clearer these structural differences are, the clearer the separation in the embedding will be. Note that this implies that shape outliers, which pose particular challenges to many previously proposed methods, will often be particularly easily detectable. Moreover, all methods for outlier detection that have been developed for tabular data inputs can be (indirectly) applied to functional data as well based on this framework, simply by using the embedding coordinates as feature inputs: The embedding space $\mathcal{Y}$ is typically a low-dimensional Euclidean space in which conventional outlier detection works well and the essential geometrical structure encoded in the pairwise functional distance matrix is conserved in these lower-dimensional embeddings. In the next section, we illustrate this practical utility in detail by extensive quantitative and qualitative analyses.

3. Experiments

To illustrate the practical relevance of the outlined geometrical approach, we first qualitatively investigate real datasets. In the second part of this section, we quantitatively investigate the anomaly detection performance of several detection methods based on synthetic data.

3.1. Qualitative Analysis of Real Data

We start with an in-depth analysis of the *ECG200* data [32,33], a functional dataset with a complex structure: it seems to contain subgroups with phase and amplitude variation and different mean functions. As a result, the dataset appears visually complex (Figure 5, left). Without the color coding, it would be challenging to identify the three subgroups (as in the lower left plot in Figure 6). Moreover, there are five left-shifted observations (apparent at $t \in [10, 25]$) and a single (partly) vertically shifted outlier (apparent at $t \in [50, 75]$), clearly detectable by the naked eye.

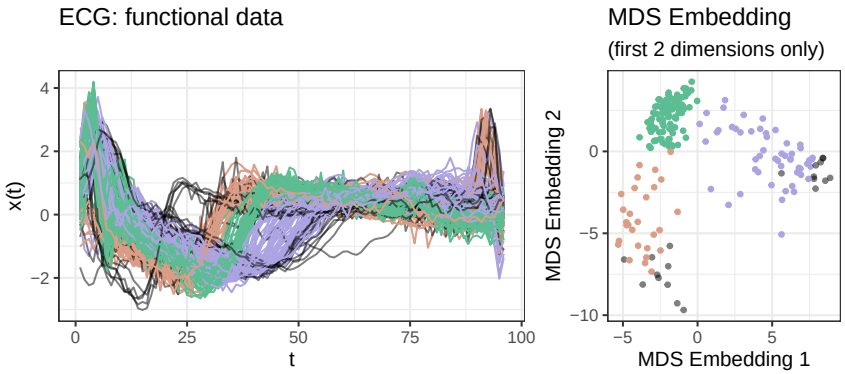

Figure 5. ECG curves and first two embedding dimensions (of five). Colors highlight subgroups apparent in the embeddings. Potential outliers with 5D-embedding LOF scores ($\texttt{minPts} = 0.75n$) in the top decile shown in black.

Much of the general structure (and the anomaly structure in particular) becomes evident in a 5D MDS embedding. To begin with, in the first two embedding dimensions, depicted on the right-hand side of Figure 5, three subgroups are easily recognizable. The color coding in Figure 5 is based on this visualization. It makes apparent that the substructures correspond to two smaller, horizontally shifted subgroups of curves (orange: left-shifted, purple: right-shifted) and a central subgroup encompassing the majority of the observations (green). In addition, we computed LOF scores on the 5D embedding

coordinates. The observations with LOF scores in the top decile are shown in black in Figure 5. This set contains all the clearly outlying observations.

More importantly, note that these observations are clearly separated from the rest in the 5D embedding shown in Figure 6: the five clearly left-shifted observations in the fourth embedding dimension and the single vertically shifted observation in the subspace spanned by the first and third embedding dimension. The figure shows a scatterplot matrix of all five embedding dimensions with observations color-coded according to the 5D-embedding LOF scores. The clearly left-shifted outliers obtain the highest LOF scores due their isolation in the subspaces including the fourth embedding dimension. Note, moreover, that other observations with higher LOF scores appear in peripheral regions of the different subspaces, but they are not as clearly separable as the six observations described before. Regarding Figure 7A, which shows the 20 most outlying curves according to LOF scores, this can be explained by the fact that these other observations stem from one of the two shifted subgroups and can thus be seen as on-manifold outliers, whereas the six other, visually clearly outlying observation, are clearly off-manifold outliers.

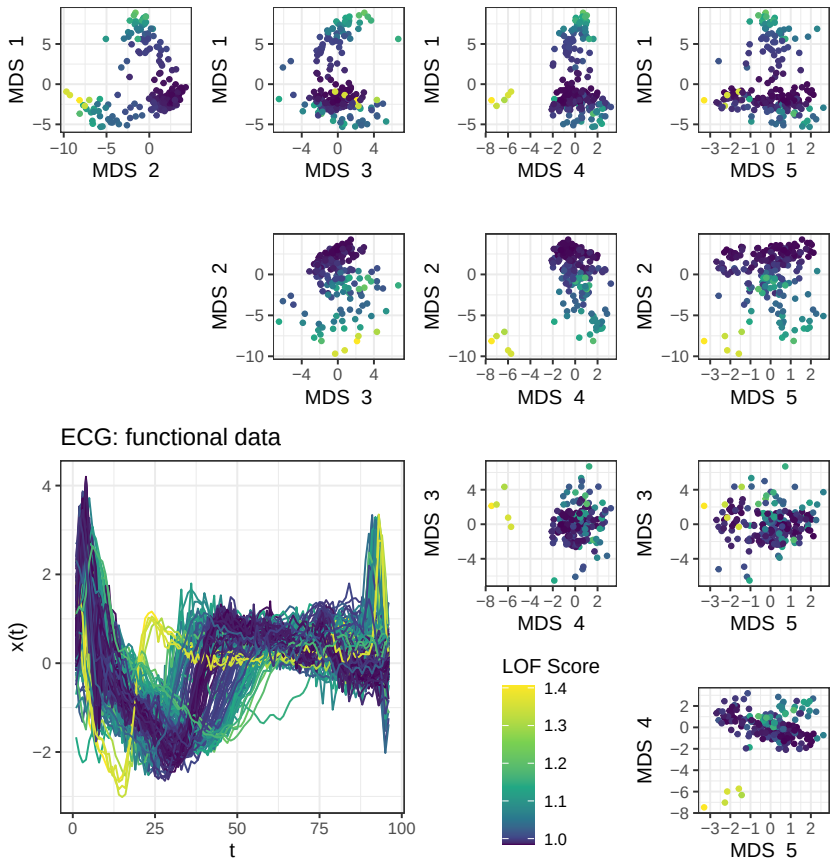

Figure 6. ECG data: scatterplot matrix of all 5 MDS embedding dimensions and curves; lighter colors for the higher LOF scores of 5D embeddings.

We contrast these findings with the results of directional outlyingness [14,34], which performs very well (see Section 3.2) on simple synthetic datasets. Figure 7 shows the ECG curves color-coded by the variation of directional outlyingness (B), the 20 most outlying curves by the variation of directional outlyingness (C), and the observations labeled as

Stats **2021**, *4* 981

outliers by directional outlyingness respectively by the MS-plot (D). First of all, it can be seen that many observations yield a high variation of directional outlyingness, and observations in the right-shifted subgroup obtain most of the highest values. In fact, among the twenty observations with the highest variation of directional outlyingness, only one is from the left-shifted group, and thirteen are from the right-shifted group. Moreover, applying directional outlyingness to this dataset results in 72 observations being labeled as outliers, which is about 36% of all observations. We would argue that it is questionable whether 36% of all observations should be labeled as outliers.

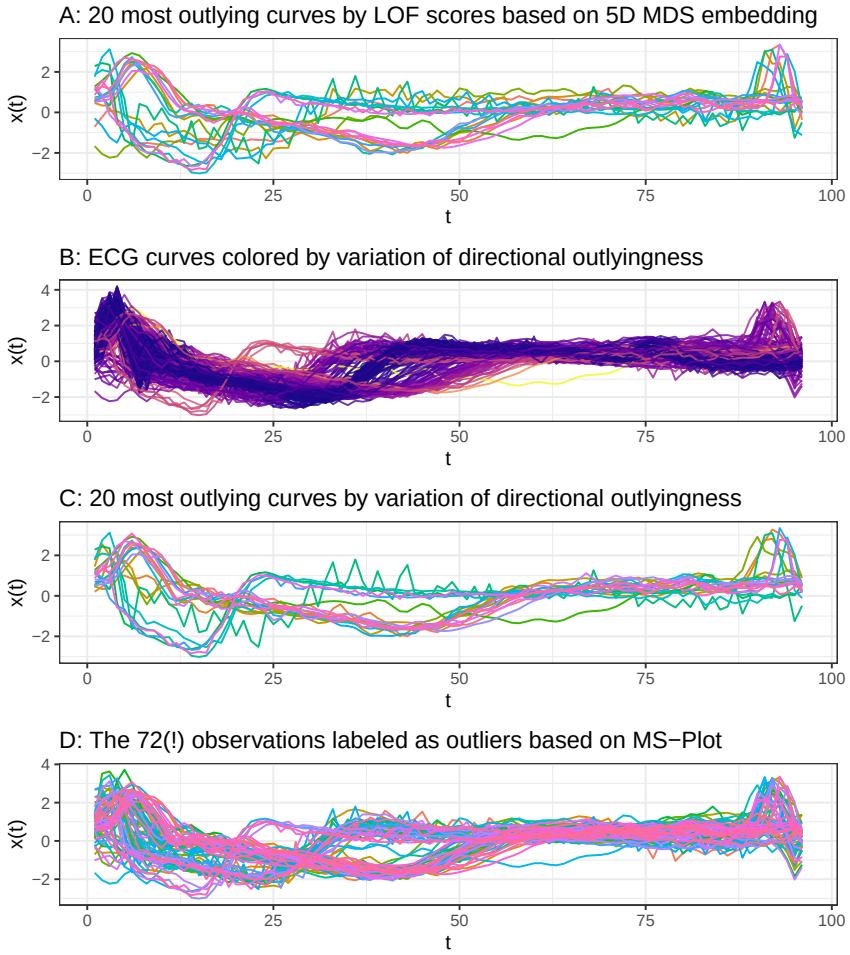

Figure 7. ECG data: LOF on MDS embeddings in contrast to directional outlyingness.

In this regard, the ECG data serve as an example that illustrates the advantages of the geometric approach. First of all, it yields readily available visualizations, which reveal much more of the inherent structure of a dataset than just its anomaly structure. This is specifically important for data with a complex structure (i.e., subgroups or multiple modes and large variability). Moreover, it allows applying well-established and powerful outlier scoring methods such as LOF to functional data. This exemplifies that the approach not only improves theoretical understanding and consideration as outlined in the previous section, it also has large practical utility in complex real data settings in which previously proposed methods may not provide useful answers.

In the ECG example, we saw that a 5D embedding yielded reasonable results and sufficiently reflected many aspects of the data. In particular, the extremely left-shifted observations became clearly separable in the fourth embedding dimension. In Appendix E, we analyze a synthetic dataset in the same way as the ECG data, which yields similar findings. Moreover, note that the Spearman rank correlation between LOF scores computed on the 5D embedding and LOF scores computed directly on the ECG data distances is 0.99. This shows that the outlier structure retained in the 5D embedding is highly consistent with the outlier structure in the high-dimensional observation space, an important aspect with respect to anomaly-scoring methods requiring (low-dimensional) tabular inputs.

Finally, note that even fewer than five embedding dimensions may suffice to reflect much of the inherent structure. Consider the examples depicted in Figure 8, which shows the functional observations and the first two embedding dimensions of a corresponding 5D MDS embedding of another four real datasets. The Octane data consist of spectra from 60 gasoline samples [35,36], the Spanish weather data of annual temperature curves of 73 weather stations [37], the Tecator data of spectrometric curves of meat samples [37,38], and the Wine data of spectrometric curves of wine samples [32,39]. As before, the observations are colored according to LOF scores based on the 5D embedding. In addition, the 12 observations with highest LOF scores are depicted as triangles. These datasets are much simpler than the ECG data, and the first two embedding dimensions already reflect the (outlier) structure fairly accurately: observations with high LOF scores appear separated in the first two embedding dimensions, and more general substructures are revealed as well. The substructure of the weather data is rather obvious already regarding the functional observations, for example, the observations with less variability in terms of temperature, all of which obtained high LOF scores. The substructure of the wine data—for example, the small cluster in the lower part of the embedding—is much harder to detect based on visualizations of the curves alone.

Appendix B summarizes a more detailed analysis of the sensitivity of the approach to the choice of the dimensionality of the embedding. We conclude that sensitivity seems to be fairly low. For all five real datasets we considered, the rank order of LOF scores is very similar or even identical whether based on two-, five-, or even twenty-dimensional embeddings (cf. Table A1).

Following Mead [40], we quantified the goodness of fit (GOF) for a d_1-dimensional MDS embedding as: $GOF(d_1) = \frac{\sum_{i=1}^{d_1} \max(0,\lambda_i)}{\sum_{j=1}^{n} \max(0,\lambda_j)}$, where λ_k are the eigenvalues (sorted in decreasing order) of the kth eigenvectors of the centered distance matrix. For all of the considered real datasets, a 5D embedding achieved a goodness of fit over 0.8, the four less-complex examples even over 0.95 (see Figure A2). As a rule of thumb, the embedding dimension does not seem crucial as long as the goodness of fit (GOF) of the embedding is over 0.8 for L_2 distances. This rule of thumb also yielded compelling quantitative performance results, as shown in Section 3.2.

Figures 6 and 8 show visualizations that combine MDS embeddings with LOF outlier scores. To put them into context, we compare them to existing visualization techniques in this section. For the sake of clarity, only the results are summarized here. The figures for the various alternative methods can be found in Appendix D. Figure A5 shows the results for the MBD-MEI "Outliergram" by Aribas-Gil and Romo [41] (implementation: [42]) for shape outlier detection and the magnitude–shape plot method of Dai and Genton [34]. Figures A6 and A7 show the results for the translation–phase–amplitude boxplots by Xie et al. [15] and the elastic depth boxplot for shape outlier detection by Harris et al. [9]. Finally, Figures A8–A13 show the corresponding functional and bivariate HDR boxplots by Hyndman and Shang [16] (implementation: [43]). Considering the MBD-MEI outliergram and the magnitude–shape plots, both of these visualization methods mostly fail to identify shift outliers (by design, in the case of the outliergram). The outliergram tends to mislabel very central observations as outliers in datasets with little shape variability (e.g., the supposed "shape outliers" detected by MBD-MEI in the central region of the Tecator data) and fails to detect even egregious shape outliers in datasets with high variability (e.g., not

a single MBD-MEI outlier in ECG 200), as well as shape outliers that are also outlying in their level (e.g., the three shape outliers identified by `msplot` in the upper region of the Tecator data). Note that some central functions of the Spanish weather data, which are labeled as outliers by the magnitude–shape plot (and partly by the outliergram) are also reflected in the 2D embedding in Figure 8.

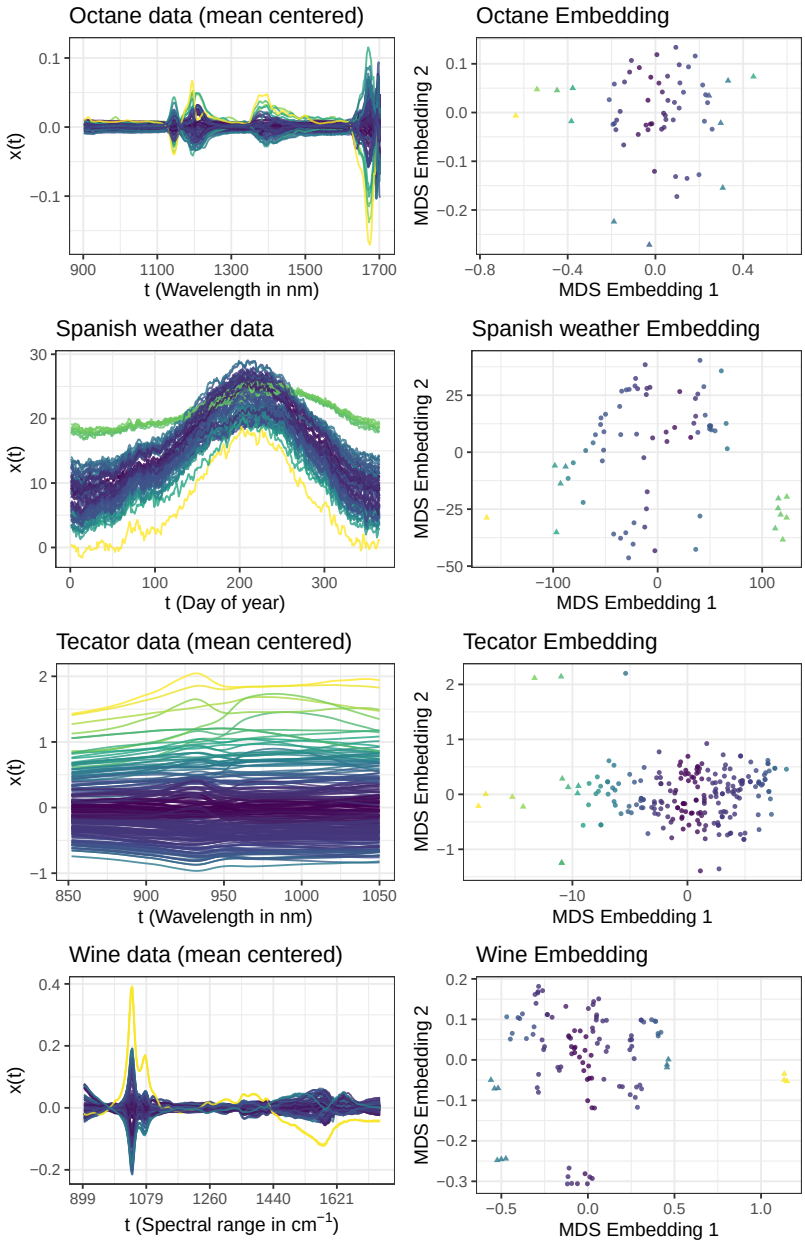

Figure 8. Further examples of real functional data colored by LOF score. The 12 most outlying observations depicted as triangles in the embedding.

They are fairly numerous relative to the overall sample size and are very similar to each other. As such, they form a clearly defined separate cluster within the data, which can be seen in the middle bottom part of the embedding. The translation–phase–amplitude boxplots mostly fail to detect outliers in data with high variability: no outliers at all are detected for the Spanish weather data despite their visually apparent anomalies, and only a single translation outlier is detected for the ECG data. Moreover, the implementation of the approach seems to break down for data with very little variation, and it was not possible to compute the phase boxplot for the Wine data, a dataset with almost no variability in terms of phase.

The results of the elastic depth boxplots do not seem to be consistent over all considered datasets. The results appear reasonable for the Octane, the Wine, and, in part, for the ECG data, where both amplitude and phase outliers are detected. However, in the ECG data, mostly observations from the left-shifted subgroup are detected as phase outliers and only two from the right-shifted subgroup. The results for the Spanish weather and the Tecator data are even less convincing. Among the Tecator data, the method labels 41 curves, i.e., 19% of all observations, as outliers, while it does not discover a single outlier in the Spanish weather data. Note, however, that the elastic depth boxplots are more robust than the translation–phase–amplitude boxplots. While the latter method only detected a single translation outlier and was not able to compute the phase boxplot for the Wine data at all, the elastic depth boxplots detect several amplitude outliers and simply do not yield phase outliers.

Finally, HDR boxplots based on PC projections of the data yield mostly similar results as the L_2-distance-based MDS embeddings. However, we would argue that dichotomizing the observations into inliers and outliers by a fixed outlier threshold makes the visualizations much less suited as an exploratory tool. Consider, for example, the Spanish weather data. The small cluster of observations with rather constant temperature ($\sim$17–25°) does not fall into the outlier region according to the dichotomization threshold, and so, they are also not shown individually in the functional HDR boxplots. Whether they are considered to be outliers or rather a subgroup surely depends on the observer, but we would argue that an outlier visualization method should emphasize and not hide such structures. Our approach of colors according to continuous scores does that very well, reflecting at the same time both the general and the outlier structure. More importantly, the outlier structure of the ECG dataset is not captured in the embedding used by the HDR boxplots. As outlined, more than two embedding dimensions are necessary to fully reflect the outlier structure of this dataset, and the density estimators underlying the HDR boxplot will break down fairly rapidly as the number of embedding dimensions increases. As such, the available implementation is limited to only using the first two PC scores for the embedding, regardless of the actual rank of the underlying data.

3.2. Quantitative Analysis of Synthetic Data

In this section, we investigate the outlier detection performance quantitatively, based on synthetic datasets for which the true (outlier) structure is known.

3.2.1. Methods

In addition to applying LOF to 5D embeddings and directly to the functional data, we investigate the performance of four "functional data"-specific outlier-detection methods: directional outlyingness (DO) [14,34], total variational depth (TV) [44], elastic depth (ED_amp, ED_pha) [9], and the approach based on translation, phase, and amplitude boxplots (AP_BOX) presented by Xie et al. [15]. For the first two methods, we use implementations provided by the package `fdaoutlier` [45] and use the variation of directional outlyingness as returned by the function `dir_out` as outlier scores for DO and the total variation depths as returned by the function `total_variation_depth` for TV. For the latter two methods, we use implementations provided by Harris et al. [9]. Outlier scores for these methods are based on elastic depths as computed by the function `depth.R1` from

the package `elasticdepth` [9] and time-warped functions as computed by the function `time_warping` from the package `fdasrvf` [46]. Note that the elastic depth approach does not produce a single outlier score per observation, but scores amplitude and phase outliers separately. Both amplitude (ED_amp) and phase (ED_pha) scores are shown in Figure 9.

3.2.2. Data-Generating Processes

The methods are applied to data from four different data-generating processes (DGPs), the first two of which are based on the simulation models introduced by Ojo et al. [25] and provided in the corresponding R package `fdaoutlier` [45]. We also provide the results of additional experiments based on the original DGPs from the package `fdaoutlier` in Appendix C. However, we consider most of these DGPs as too simple for a realistic assessment, as most methods achieve almost perfect performance on them, and we use more complex DGPs here. In both DGPs 1 and 2, the inliers from `simulation_model1` from the package `fdaoutlier` serve as $\mathcal{M}_c$, i.e., the common data-generating process. This results in simple functional observations with a positive linear trend. In addition, `simulation_model1` generates simple shift outliers. Additionally, our DGP 1 also includes shape outliers stemming from `simulation_model8`, which serves as $\mathcal{M}_a$. In contrast, DGP 2 contains shape outliers from all of the other DGPs in `fdaoutlier`, which means $\mathcal{M}_a$ contains observations from several different data-generating processes.

For DGPs 3 and 4, we define $\mathcal{M}_c$ by generating a random, wiggly template function over $[0, 1]$ for each dataset, generated from a B-spline basis with 15 or 25 basis functions, respectively, with i.i.d. $\mathcal{N}(0, 1)$ spline coefficients. Functions in $\mathcal{M}_c$ are generated as elastically deformed versions of this template, with random warping functions drawn from the ECDFs of Beta(a, b) distributions with $a, b \sim U[4, 6]$ (DGP 3) or $a, b \sim U[3, 8]$ (DGP 4). Functions in $\mathcal{M}_a$ are also generated as elastically deformed versions of this template, with Beta ECDF random warping functions with $a, b \sim U[3, 4]$ for DGP 3 and with 50:50 Beta mixture ECDF random warping functions with $a, b \sim 0.5U[3, 8] : 0.5U[0.1, 3]$ (DGP 4). Finally, white noise with $\sigma = 0.1, 0.15$, respectively, for DGPs 3 and 4 is added to all resulting functions. Appendix F shows visualizations of example datasets drawn from these DGPs.

3.2.3. Performance Assessment

From these four DGPs, we sampled data $B = 500$ times with three different outlier ratios $r \in \{0.1, 0.05, 0.01\}$. Based on the outlier scores, we computed the area under the ROC curve (AUC) and Mathew's correlation coefficient (MCC) as the performance measures and report the results over all 500 replications. Note that, for $r \in \{0.1, 0.05\}$, the number of sampled observations was $n = 100$, whereas for $r = 0.01$, we sampled $n = 1000$ observations. Since computing the elastic depths and time-warped functions requires more than an hour for a single dataset with 1000 observations, we only included them for the settings with 100 observations.

3.2.4. Results

We note that LOF applied directly to functional data distances yielded very similar results as LOF applied to their 5D embeddings. This agrees with our findings in the qualitative analyses. In the following, we simply refer to the geometrical approach and do not distinguish between the LOF based on MDS embeddings and the LOF applied directly to the functional distance matrix. Figure 9 shows that the proposed geometrical approach is highly competitive with existing functional-data-specific outlier-detection methods. It yields better results than TV for all of the four DGPs and performs at least on par with DO. In comparison to DO, it performs better on DGP 1 and DGP 3, on par on DGP 4, and worse on DGP 2. Note that DO struggles to detect simple shift outliers: among these methods, it performs worst on the first DGP. Similar conclusions can be reported for other settings, where it performs even worse if there are only shift outliers (cf. Figures A3 and A15). Moreover, while the approaches based on elastic depth proposed by Harris et al. (ED_amp

and ED_pha) and the approach proposed by Xie et al. (AP_BOX) perform well on DGP 2, they are outperformed by DO in this setting, and on DGPs 1, 3, and 4, they clearly perform the worst. Thus, these two methods yield the worst performances overall.

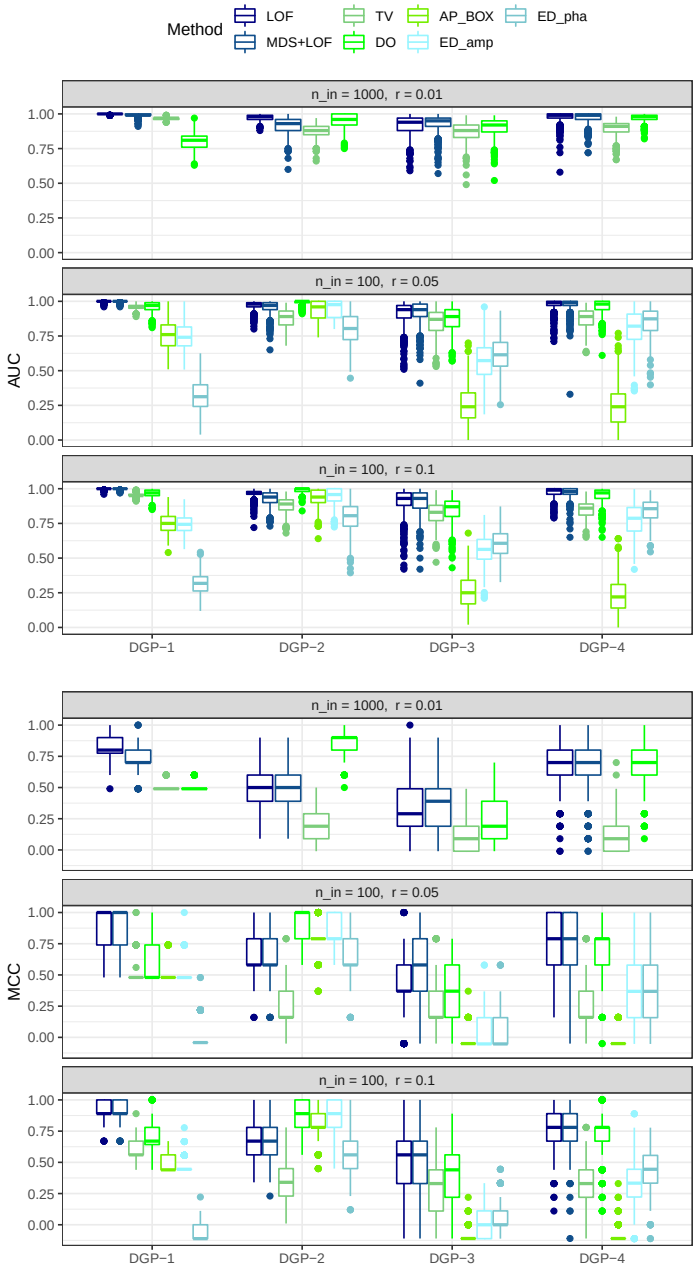

Figure 9. Distribution of the AUC and MCC over the 500 replications for the different data-generating processes (DGPs), outlier-detection methods, and outlier ratios r.

Note that the insights we gain on synthetic data are confirmed by all of the real data applications we investigate in Section 3.1. In addition to the experiments conducted here, we applied the considered methods and their accompanying visualization approaches to these five real datasets. The results of the previously proposed visualizations are presented in detail in Appendix D, Figures A5–A7. In contrast to the proposed geometrical approach, none of them yields satisfactory results consistently for all of the considered datasets. For example, the outliergram, as well as the approach based on translation, phase and amplitude boxplots and the elastic depth approach fail to identify any outliers in some of these datasets, while the magnitude–shape plot, for example, labels an entire third of all observations in the ECG data as outliers (as already outlined in Section 3.1).

In summary, based on the conducted experiments, the proposed geometrical approach yields very compelling results: On synthetic data, it leads to outlier scoring performances at least on par with specialized functional-outlier-detection methods even in its simplest version (MDS with L_2 distances and LOF). Moreover, in contrast to the other methods, it yields consistently useful and sensible results on all of the considered real datasets, while providing more intuitive and more easily interpretable visualizations. Going further, our approach can be adapted to specific settings simply by choosing metrics other than L_2. As the next section shows, this can improve the outlier-detection performance considerably.

3.3. General Dissimilarity Measures and Manifold Methods

So far, we have computed MDS embeddings mostly based on L_2 distances. In the following, we show that the approach is more general. The geometric structure of a dataset is captured in the matrix of pairwise distances among observations. Different metrics emphasize different aspects of differences in the data and can thus lead to different geometries. MDS based on L_2 distances yielded compelling results in many of the examples considered above, but other distances are likely to lead to better performance in certain settings. To illustrate the effect, we consider two additional settings—one simulated and one on real data—in the following. The results are displayed in Figure 10.

The simulated setting is based on isolated outliers, i.e., observations that deviated from functions in $\mathcal{M}_c$ only on small parts of their domain. In such settings, higher-order L_p metrics lead to better results, since such metrics amplify the contribution of small segments with large differences to the total distance. We use as an example data generated from `simulation_model2` from the package `fdaoutlier`. Figure 10A shows the AUC values of LOF scores on MDS embeddings based on L_2 and L_{10} distances. Again, 500 datasets were generated form the model over different outlier ratios. In contrast to L_2-based MDS, using L_{10} distances yielded almost perfect detection. In embeddings based on L_{10}, isolated outliers are clearly separable in the first two or three embedding dimensions.

As a second example, we consider the *ArrowHead* dataset [47,48], which contains outlines of three different types of neolithic arrowheads (see Appendix G for visualizations of the dataset). Using the 78 structurally similar observations from class "Avonlea" as our data on $\mathcal{M}_c$ and sampling outliers from the 126 structurally similar observations from the other two classes, we can compute AUC values based on the given class labels. We generate 500 datasets for each outlier ratio $r \in \{0.05, 0.1\}$. Since there are only 78 observations in the class "Avonlea", we do not use $r = 0.01$ for this example. Embeddings are computed using three different dissimilarity measures: the standard L_2 metric, the unnormalized L_1-Wasserstein metric [31], and the dynamic time warping (DTW) distance [49]. Note that the DTW distance does not define a proper metric [50].

Figure 10B shows that small performance improvements can be achieved in this case if one uses dissimilarity measures that are more appropriate for the comparison of shapes, but not as much as in the isolated outlier example. Note that even though the DTW distance is not a proper metric, it improves the outlier-scoring performance in this example. This indicates that, from a practical perspective, general dissimilarity measures can be sufficient for our approach to work. This opens up further possibilities, as there are many general dissimilarity measures for functional data, for example the semimetrics introduced by

Fuchs et al. [51]. Overall, these examples illustrate the generality of the approach: using suitable dissimilarity measures can make the respective structural differences more easily distinguishable.

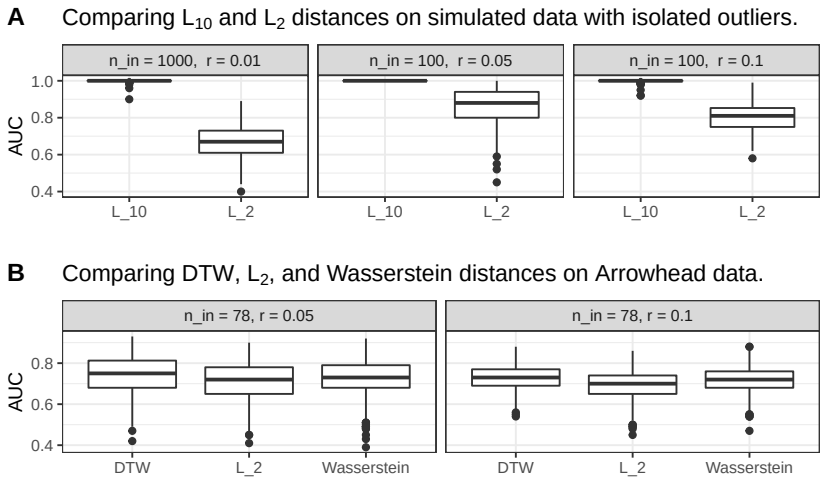

Figure 10. Comparing the effects of different distance measures. Depicted are the distributions of the AUC over 500 replications for the LOF based on MDS embeddings computed with the respective distance measures for different outlier ratios r. (**A**) Comparing the L_{10} and L_2 metrics on a dataset with isolated outliers generated via Simulation Model 2 from the package fdaoutlier. (**B**) Comparing the DTW, L_2, and unnormalized L_1-Wasserstein distance measures on the real dataset ArrowHead. Note: the DTW distance is not a metric.

More complex embedding methods, on the other hand, do not necessarily lead to better or even comparable results as MDS. Figure 11 shows the distribution of the AUC for embedding methods ISOMAP and UMAP. Both methods require a parameter that controls the neighborhood size used to construct a nearest neighbor graph from which the manifold structure of the data is inferred. The larger this value, the more of the global structure is retained. For both methods, embeddings were computed for very small and very large neighborhood sizes of five and ninety.

The results show that neither method performs better than MDS; UMAP even performs considerably worse. Note that ISOMAP is equivalent to MDS based on the geodesic distances derived from the nearest neighbor graph, and the larger the neighborhood size the more similar to direct pairwise distances these geodesic distances become. This is also reflected in the results, as ISOMAP-90 performs better than ISOMAP-5 on average. For DGP-2, ISOMAP-90 slightly outperforms MDS, indicating that more complex manifold methods could improve the results somewhat in specific settings.

In general, however, these findings confirm the theoretical considerations sketched in Section 2.2. Embedding methods that preserve the geometry of the space $\mathcal{F}$ of which $\mathcal{M}_c$ and $\mathcal{M}_a$ are submanifolds, i.e., the ambient space geometry, are more suited for outlier detection than methods that focus on approximating the intrinsic geometry of the manifold(s). Thus, more sophisticated embedding methods, which often focus on approximating the intrinsic geometry, should not be applied lightly and certainly require careful parameter selection in order to be applicable for outlier detection. Since hyperparameter tuning for unsupervised methods remains an unsolved problem, this is unlikely to be achieved in real-world applications. In particular, consider that both UMAP and t-SNE [29] have been found to be—in general—oblivious to local density, which means that clusters of different density in the observation space tend to become clusters of more equal density in the embedding space [52]. Although there may exist a parameter setting where this effect is

Stats **2021**, *4* 989

reduced (note that there are now density-preserving versions of t-SNE and and UMAP [52]), we are skeptical that outliers can be faithfully represented in such an embedding given the difficulties of hyperparameter tuning in unsupervised settings. Moreover, these methods are not designed to preserve important aspects of the outlier structure. For example, UMAP is subject to a local connectivity constraint, which ensures that every observation is at least connected to its nearest neighbor (in more technical terms: that a vertex in the fuzzy graph approximating the manifold is connected by at least one edge with an edge weight equal to one [30]), which makes it unlikely that UMAP can be tuned so that it is able to sensibly embed off-manifold outliers, which should, by definition, not be connected to the common data manifold. The poor performance of UMAP embeddings in our experiments confirms these concerns.

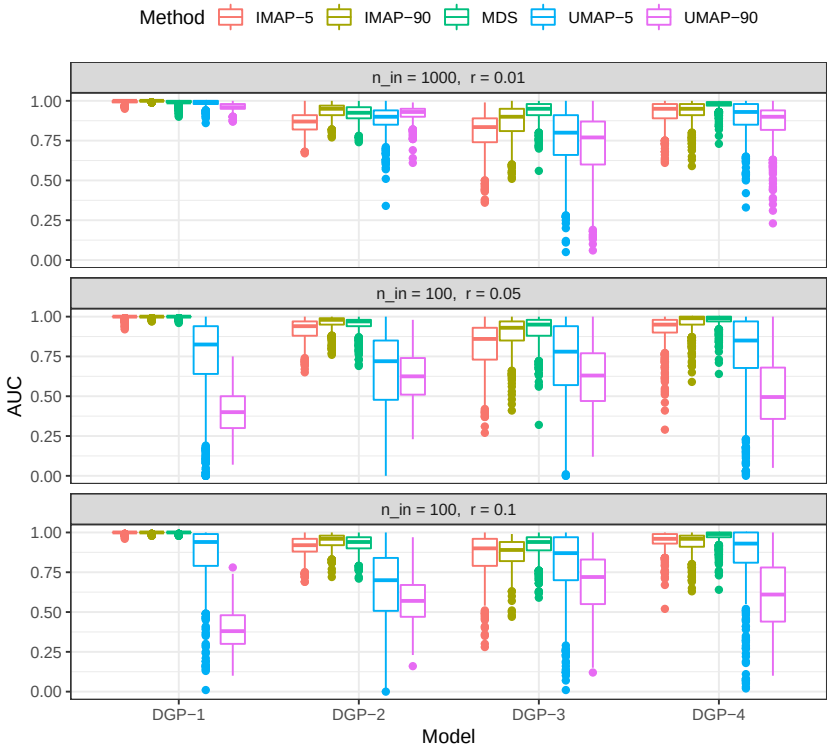

Figure 11. Comparing UMAP and ISOMAP to MDS. UMAP and ISOMAP embeddings were computed for two different locality parameter values: 5 and 90. The distribution of the AUC over 500 replications of the four DGPs for different outlier ratios r. The AUC computed on LOF scores based on 5D embeddings.

4. Discussion

Based on a geometrical perspective of functional outlier detection, we defined two general types of functional outliers: off- and on-manifold outliers. Our investigation showed that this perspective clarifies the theoretical concepts and improves practical results. From a theoretical perspective, it allows formalizing functional outlier scenarios in precise and consistent terms, beyond differences in terms of either shape, level, or magnitude. This simplifies reasoning about specific outlier settings and provides a fully general theoretical conceptualization of the problem.

From an applied perspective, we formulated two important consequences. First of all, as was demonstrated with a comprehensive analysis of a complex, real dataset of

ECG curves, the geometrical approach allows for easily accessible and highly informative visualizations. These are obtained by means of low-dimensional embeddings reflecting the inherent structure of a functional dataset in much detail. Such visualizations provide more accurate and complete pictures of the (outlier) structure of functional data. In particular, off-manifold outliers reliably appear as clearly separated (groups of) points in the low-dimensional embeddings.

Second, the proposed approach makes it possible to apply highly developed and performant standard outlier-detection methods to functional data, since the geometric structure of the data is captured and reflected in their pairwise distance matrices. Outlier detection and scoring methods that can be applied to distance matrices can therefore directly be used for functional data as well. Furthermore, detection methods requiring tabular inputs can also be applied simply by using the embedding coordinates obtained with embedding methods as proxy data for the original functions. Our experiments using LOF scores showed that the two approaches yielded very similar results. This simultaneously simplifies and improves functional outlier detection: It simplifies since functional data analysis becomes more accessible to a broader audience with general outlier-detection methods that are widely used in other areas and that do not require an understanding of complex methodological details of functional data methods. It improves the state-of-the-art since many functional outlier methods can only detect specific kinds of functional outliers by design or fail in more complex realistic data that are widely dispersed or that contain multiple nonoutlying subgroups, such as the ECG data. Moreover, note that our proposal is not limited to univariate functional data. Extending it to multivariate functions is completely straightforward, as long as a suitable dissimilarity measure is available to compute pairwise distances.

In this paper, most embeddings were obtained using MDS based on L_2 distances. This implies a close similarity to functional bagplots and highest-density region (HDR) boxplots [16], which are based on the first two robust principal component scores. However, this similarity only applies if our geometrical approach is implemented with 2D MDS embeddings based on L_2 distances. As outlined, our proposal is neither limited to the L_2 metric as a distance measure nor to MDS as an embedding method or just two embedding dimensions. Other metrics and (higher-dimensional) embedding methods can be used as well, and our results indicate that an alternative distance measure can further improve the performance in specific settings, sometimes considerably. In particular, even nonmetric dissimilarity measures may be applicable as our results based on DTW distances indicate. On the other hand, the results also show that more sophisticated embedding methods such as ISOMAP and UMAP cannot be used as straightforwardly as MDS. Such methods, which do not take into account the ambient space geometry by default, at least require very careful parameter selection.

In terms of practical applicability, the $O(n^3)$ time complexity and $O(n^2)$ storage complexity of standard MDS may prove problematic for large data, but generalizations such as Landmark MDS [53], Pivot MDS [54], or multilevel MDS exploiting GPU performance [55] scale much better with the number of available observations.

Finally, we would argue that existing functional outlier detection approaches mostly lack the principled geometrical underpinning and conceptualization presented here. As outlined, we argue that such a conceptualization is necessary to make functional outlier detection tractable in full generality. Specifically, consider that existing methods typically limit themselves to creating a 1D or 2D representation of each curve (e.g., MBD-MEI, MO-VO, functional bagplots, HDR plots), often based on preconceived notions of the characteristics of functional outliers. Our investigations and experiments suggested that this is often not sufficient for real-world functional outlier detection: there is no valid reason to limit our representations to two dimensions with modern outlier-detection methods, and the geometrical perspective often strongly suggests otherwise in the case of complex functional data. Even more importantly, it is much more flexible to learn maximally informative low-dimensional representations directly from data instead of starting with

rigid notions of which characteristics to look at and to ignore the rest. The latter is likely to lead to results not capturing the entire (outlier) structure of a given dataset, which is essential in real-world unsupervised settings and exploratory analyses.

Based on the theoretical considerations and the empirical results outlined above, we conclude that the proposed approach is well suited for both the theoretical conceptualization and the practical implementation of functional outlier detection. In particular, the choice of embedding method should consider whether it is able to preserve the extrinsic geometry of the function space, and simple MDS embeddings based on functional distances provide a very strong baseline for that. On the basis of this work, we intend to further investigate the implications of the geometrical perspective, such as the effects other dissimilarity measures, embedding, and outlier-detection methods, in future research. We are also investigating the use of mass volume curves [56] for hyperparameter tuning in functional outlier detection. Such a criterion will permit analysts to optimize the combination of the functional distance metric, embedding dimensionality, and outlier-scoring method parameters. In the absence of quantitative criteria for optimizing these settings, our recommendations are to (1) use the standard L_2 metric as the default, which proved to be a very strong baseline in our experiments for a wide variety of data settings and outlier types, (2) make use of substantive knowledge about the data at hand, either from an initial exploratory data analysis or expertise about the data-generating process, in order to choose metrics that are sensitive to the relevant kinds of structural deviations, and (3) supplement and verify the results with results based on alternative metrics, since our proposal has a low computational cost for typical functional dataset sizes.

Author Contributions: Conceptualization, M.H.; methodology, M.H. and F.S.; software, M.H.; validation, M.H. and F.S.; formal analysis, M.H. and F.S.; investigation, M.H. and F.S.; resources, M.H. and F.S.; data curation, M.H.; writing—original draft preparation, M.H.; writing—review and editing, M.H. and F.S.; visualization, M.H. and F.S.; supervision, F.S.; project administration, F.S.; funding acquisition, F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibility for its content.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All R code and data to fully reproduce the results are freely available on GitHub: https://github.com/HerrMo/fda-geo-out.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

LOF	Local outlier factor
FD(A)	Functional data (analysis)
(F)PCA	(Functional) principle component analysis
HDR	High-density region
NHST	Null hypothesis significance testing
ECG	Electrocardiogram
MDS	Multidimensional scaling
DTW	Dynamic time warping
MS-plot	Magnitude–shape plot
GOF	Goodness of fit
DO	Directional outlyingness
TV	Total variational depth
ED	Elastic depth
DGP	Data-generating process
ECDF	Empirical cumulative distribution function
AUC	Area under the ROC curve

MBD	Modified band depth
MEI	Modified epigraph index
MO	Mean directional outlyingness
VO	Variability of directional outlyingness

Appendix A. Formalizing Phase Variation Scenarios

Appendix A.1. Phase Variation: Case I

The manifold $\mathcal{M} = \{x(t) : x(t) = \theta_1 \varphi(t - \theta_2), \theta = (\theta_1, \theta_2)' \in \Theta\}$, with $\varphi(.)$ the standard Gaussian pdf and $\Theta = [0.1, 2] \times [-2, 2]$, defines a functional data setting with independent amplitude and phase variation. Since there is a single manifold only, there are no structural novelties. Figure A1, top, depicts the functional observations on the left and a 2D embedding obtained with MDS on the right. Note that all of the curves are subject to amplitude and phase variation to a varying extent; however, there are no clearly "outlying" or "outstanding" observations in terms of either amplitude or phase. This is reflected in the corresponding embedding, which does not show any clearly separated observations in the embedding space, indicating that there are no structurally different observations. The situation in the second case of phase-varying data, however, is different.

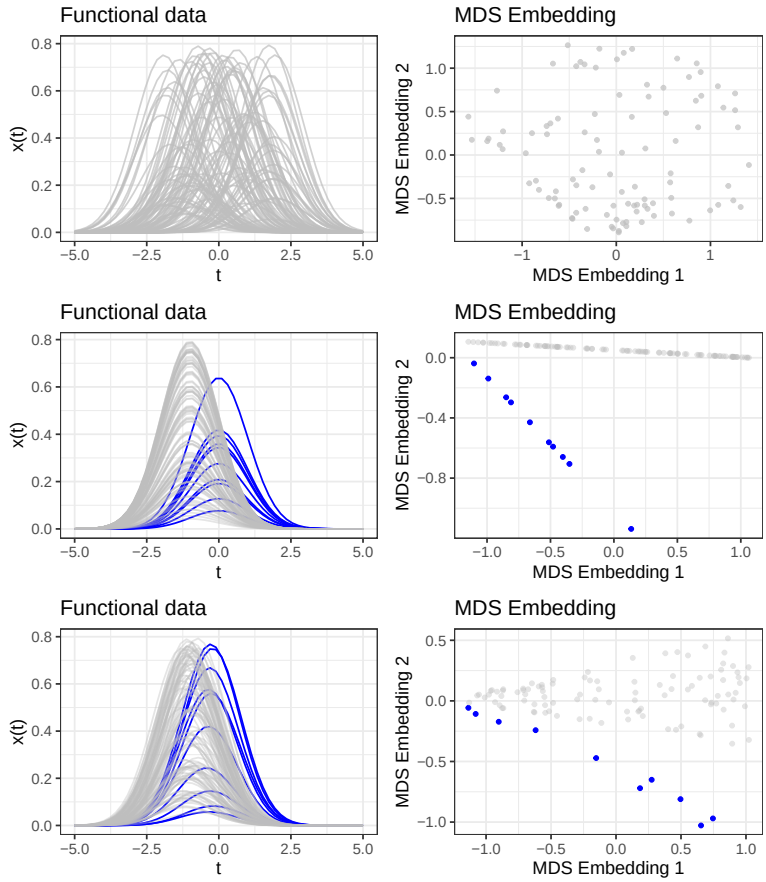

Figure A1. Functional data with phase variation and different levels of structural difference. **Top**: scenario with no off-manifold outliers. **Middle**: scenario with clear off-manifold outliers. **Bottom**: intermediate scenario.

Appendix A.2. Phase Variation: Case II

The two manifolds $\mathcal{M}_c = \{x(t) : x(t) = \theta\varphi(t+1), \theta \in \Theta\}$ and $\mathcal{M}_a = \{x(t) : x(t) = \theta\varphi(t), \theta \in \Theta\}$, with $\Theta = [0.1, 2]$ describe a similar scenario as before; however, there are two structurally different manifolds induced by the shift in the argument of φ. In contrast to the first case, there are on-manifold and off-manifold outliers. Figure A1, middle, depicts the functional observations and the corresponding embedding. Clearly, in this example, a few (blue) curves, the ones from $\mathcal{M}_a$, show a horizontal shift compared to the normal data, and consequently, those few curves appear horizontally "outlying". Within the main data manifold, only on-manifold outliers in terms of amplitude exist. These aspects are reflected in the corresponding embedding: the low-dimensional representations of the blue curves are clearly separated from those of the main data in grey.

Of course, such clear settings—in particular, phase-varying functional data with fixed and distinct phase parameters—will seldom be observed in practice. A more realistic example is given by $\mathcal{M}_c = \{x(t) : x(t) = \theta_1\varphi(t-\theta_2), (\theta_1, \theta_2)' \in \Theta_c\}$ and $\mathcal{M}_a = \{x(t) : x(t) = \theta_1\varphi(t-\theta_2), (\theta_1, \theta_2)' \in \Theta_a\}$, with $\Theta_c = [0.1, 2] \times [-1.3, -0.7]$ and $\Theta_a = [0.1, 2] \times [-0.5, 0.1]$. Here, we have again two structurally different manifolds. This is more realistic, since the "phase parameters" θ_2 are not fixed, but are subject to random fluctuations. In addition, the structural difference induced by the phase parameters is much smaller. Considering Figure A1, bottom, again, this is reflected in the embedding: there are two separable structures; however, the differences are not as clear as in the second example above.

The three examples together show that the less similar the processes are and/or the less variability there is within the phase parameters defining the manifolds, the clearer structural differences induced by horizontal variation become visible in the embeddings.

Appendix B. Sensitivity Analysis

The differences in complexity among the ECG and the other four real datasets become apparent in Figure A2 as well, which shows how the goodness of fit (GOF) of the embeddings is affected by their dimensionality. For the L_2 metric, a goodness of fit over 0.9 is achieved with two to three embedding dimensions for the less complex datasets. Moreover, all of them reach a saturation point at five dimensions. This is in contrast to the ECG data, where the first five embedding dimensions lead to a goodness of fit of 0.8. Moreover, the ranking induced by LOF scores is very robust to the number of embedding dimensions. As Table A1 shows, the rank correlations between LOF scores based on five and LOF scores based on twenty embedding dimensions are very high for all datasets.

Table A1. Spearman correlation between LOF scores based on embeddings of different dimensionality for the 5 considered real datasets and metrics $L_{0.5}$, L_1, ..., L_{10}, and unnormalized L_1-Wasserstein. MDS embeddings with 5 dimensions are compared to embeddings with 2 (2 vs. 5) and 20 (5 vs. 20) dimensions.

	$L_{0.5}$		L_1		L_2		L_3		L_4		L_5	
	2 vs. 5	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**
ECG	0.96	0.97	0.98	0.97	0.97	0.99	0.94	0.99	0.94	0.98	0.90	0.97
Octane	0.94	0.99	0.96	0.98	0.97	0.99	0.98	0.99	0.98	0.99	0.96	0.98
Weather	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Tecator	0.97	0.99	0.96	0.99	0.99	1.00	0.99	1.00	0.99	1.00	1.00	1.00
Wine	0.98	0.99	0.99	1.00	1.00	1.00	1.00	1.00	0.99	1.00	0.99	1.00
	L_6		L_7		L_8		L_9		L_{10}		**Wasserstein**	
	2 vs. 5	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**	**2 vs. 5**	**5 vs. 20**
ECG	0.89	0.96	0.87	0.96	0.86	0.95	0.86	0.95	0.85	0.94	0.98	0.97
Octane	0.96	0.98	0.95	0.99	0.96	0.98	0.94	0.97	0.94	0.97	0.95	0.96
Weather	1.00	1.00	0.99	1.00	0.99	1.00	0.99	1.00	1.00	1.00	1.00	1.00
Tecator	1.00	1.00	1.00	1.00	0.99	1.00	0.99	1.00	0.99	1.00	0.96	0.99
Wine	0.99	1.00	0.98	1.00	0.98	1.00	0.98	0.99	0.98	0.99	0.99	1.00

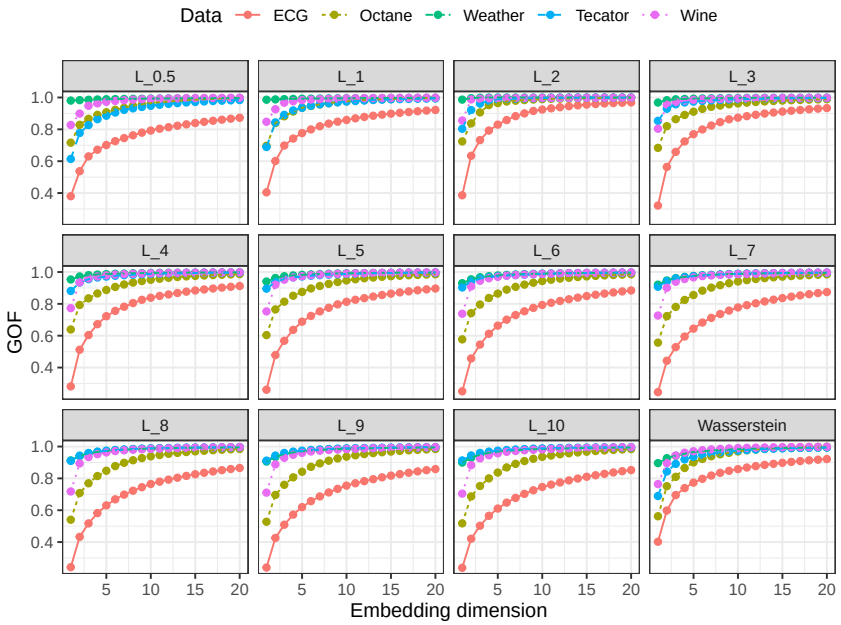

Figure A2. Goodness of fit (GOF) of different embedding dimensions for the five considered real datasets and $L_{0.5}, L_1, ..., L_{10}$, and unnormalized L_1-Wasserstein metrics.

Appendix C. Quantitative Results on the `fdaoutlier` Package DGPs

The simulation models presented by Ojo et al. [25] cover different outlier scenarios: vertical shifts (Model 1), isolated outliers (Model 2), partial magnitude outliers (Model 3), phase outliers (Model 4), various kinds of shape outliers (Models 5–8), and amplitude outliers (Model 9). A detailed description can be found in the vignette (https://cran.r-project.org/web/packages/fdaoutlier/vignettes/simulation_models.html, accessed on 15 November 2021) accompanying their R package. In the following, the proposed geometrical approach is compared to directional outlyingness (DO) and total variational depth (TV) using the AUC as a performance measure.

As Figure A3 shows, (almost) perfect performance is achieved by at least two methods for Models 1, 3, 4, 8, and 9; DO shows almost perfect performance for all models except Model 1. For Models 2, 5, 6, and 7, the methods based on the geometric approaches do not perform equally well (as does TV). However, as outlined in Section 3.3, perfect performance can be achieved for Model 2 by using L_{10} distances instead of L_2 distances.

Furthermore, for Models 5, 6, and 7, it has to be taken into account that the AUC values only reflect the detection of "true outliers", which can now—given the geometric perspective—be specified more precisely as off-manifold outliers (observations from $\mathcal{M}_a$). However, this does not take into account possible on-manifold outliers. Due to their distributional nature, by chance, some on-manifold outliers (observations on $\mathcal{M}_a$) can be "more outlying" than some of the off-manifold outliers and thus correctly obtain higher LOF scores. However, such cases are not correctly reflected in the performance assessment approach, as—in contrast to off-manifold outliers—such on-manifold outliers are not labeled as "true outliers". The observed lower performance in terms of the AUC thus can simply mean that there are on-manifold outliers obtaining relatively high LOF scores. In particular, this also does not imply that off-manifold outliers fail to be separated in a subspace of the embedding, as will be outlined in Appendix E in more detail, nor that perfect AUC performance cannot be obtained via the geometric approaches for these settings. If the geometric approach is applied to the derivatives instead (depicted in

Figure A3 as "deriv"), almost perfect performances can be achieved. Obviously, functions of the same shape (i.e., all observations from $\mathcal{M}_c$) are very similar on the level of derivatives regardless of how strongly dispersed they are in terms of vertical shift.

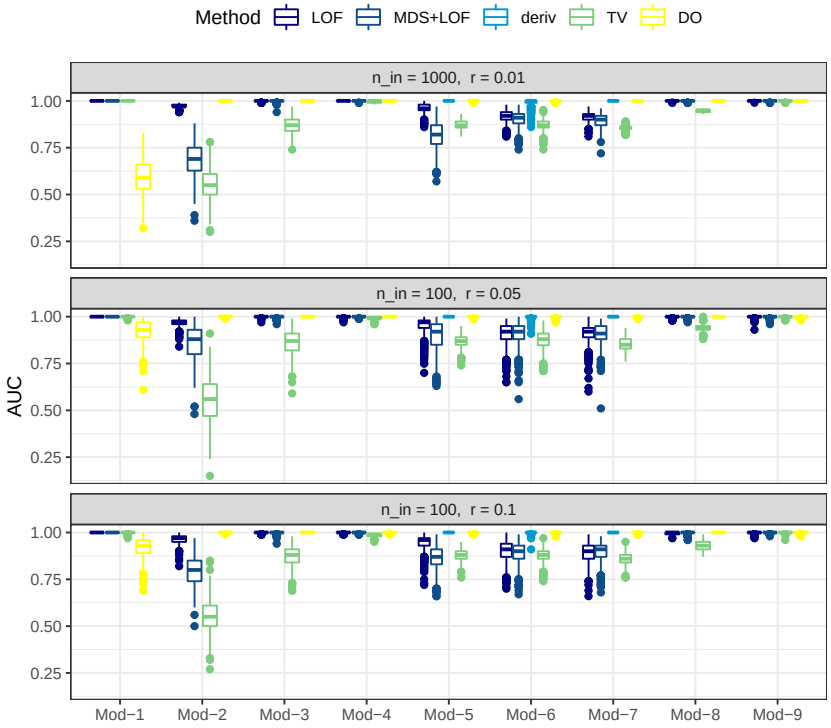

Figure A3. Distribution of the AUC over the 500 replications for the different outlier-detection methods, simulation models (Mod) from the package fdaoutier, and outlier ratios r.

Appendix D. Visualization Methods: `roahd::outliergram`, `fdaoutlier::msplot`, Translation–Phase–Amplitude Boxplots, Elastic Depth Boxplots, and HDR Boxplots

Figure A4 shows the results for the synthetic data example of Figure 4 with ten true outliers, where the MS plot yields six false positives and only three true positives, while the Outliergram fails to detect even a single outlier. The elastic depth boxplots labels twenty-six observations as outliers, only two of which are among the shifted observations. Moreover, note that observations labeled phase outliers are also labeled amplitude outliers at the same time. In contrast, the translation–phase–amplitude boxplots correctly detect the 10 shifted observations as translation outliers; however, 15 other observations are also labeled outliers. Note that some observations obtain multiple labels, for example, all phase outliers are also labeled as amplitude outliers. The HDR boxplots yield six false positives and no true positive (see Figure A13). In summary, neither of the methods are capable of correctly capturing the outlier structure of this dataset, in contrast to the proposed geometrical approach.

Figure A5 shows results for the MBD-MEI "Outliergram" by Aribas-Gil and Romo [41] (implementation: [42]) for shape outlier detection, and the magnitude–shape plot method of Dai and Genton [34] for the example datasets shown in Figures 5 and 8. Figures A6 and A7 show the results for the translation–phase–amplitude boxplots by Xie et al. [15] and the elastic depth boxplot for shape outlier detection by Harris et al. [9] for these datasets. Finally, Figures A8–A13 show the results of the HDR boxplots by Hyndman and Shang [16] (implementation: [43]). For a detailed discussion, see Section 3.1.

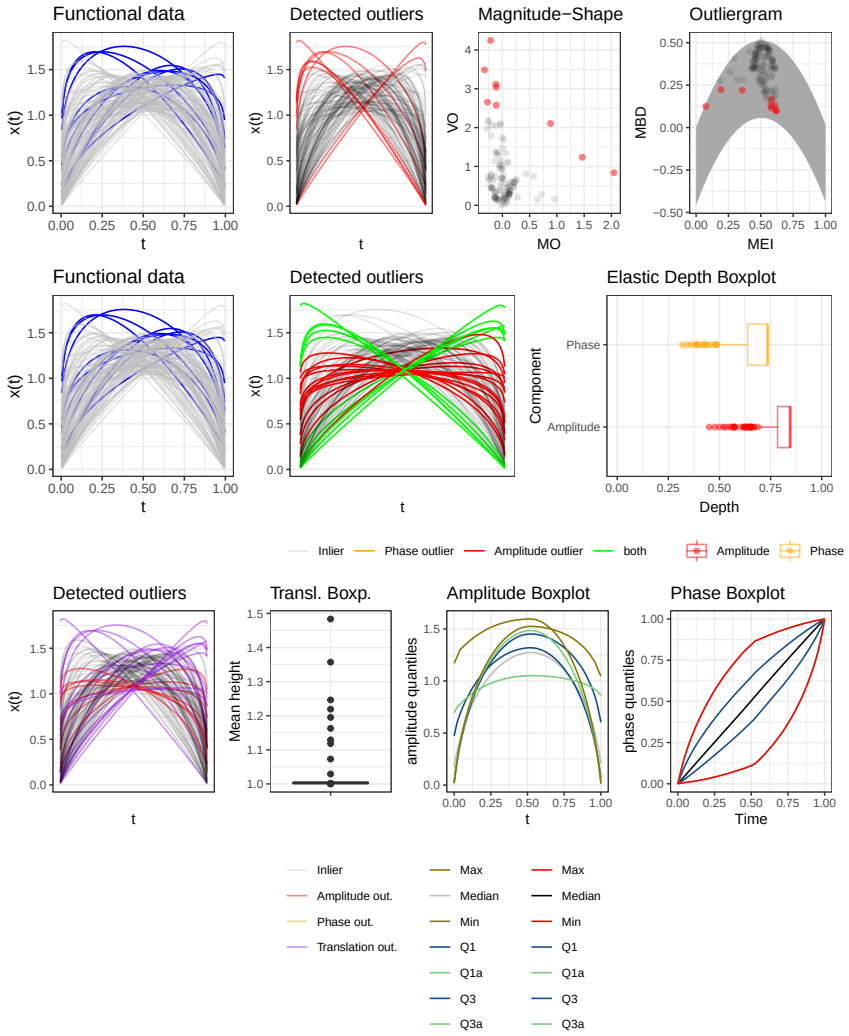

Figure A4. First column of first two rows: data with true outliers in blue; subsequent columns: data with detected outliers in color. First row: magnitude–shape plot of mean directional outlyingness (MO) versus variability of directional outlyingness (VO) and outliergram of the modified epigraph index (MEI) versus modified band depth (MBD) with the inlier region in grey. Second row: Elastic depth boxplots. Third row: translation–phase–amplitude boxplots. For the results of the HDR boxplots on the data, see Figure A8.

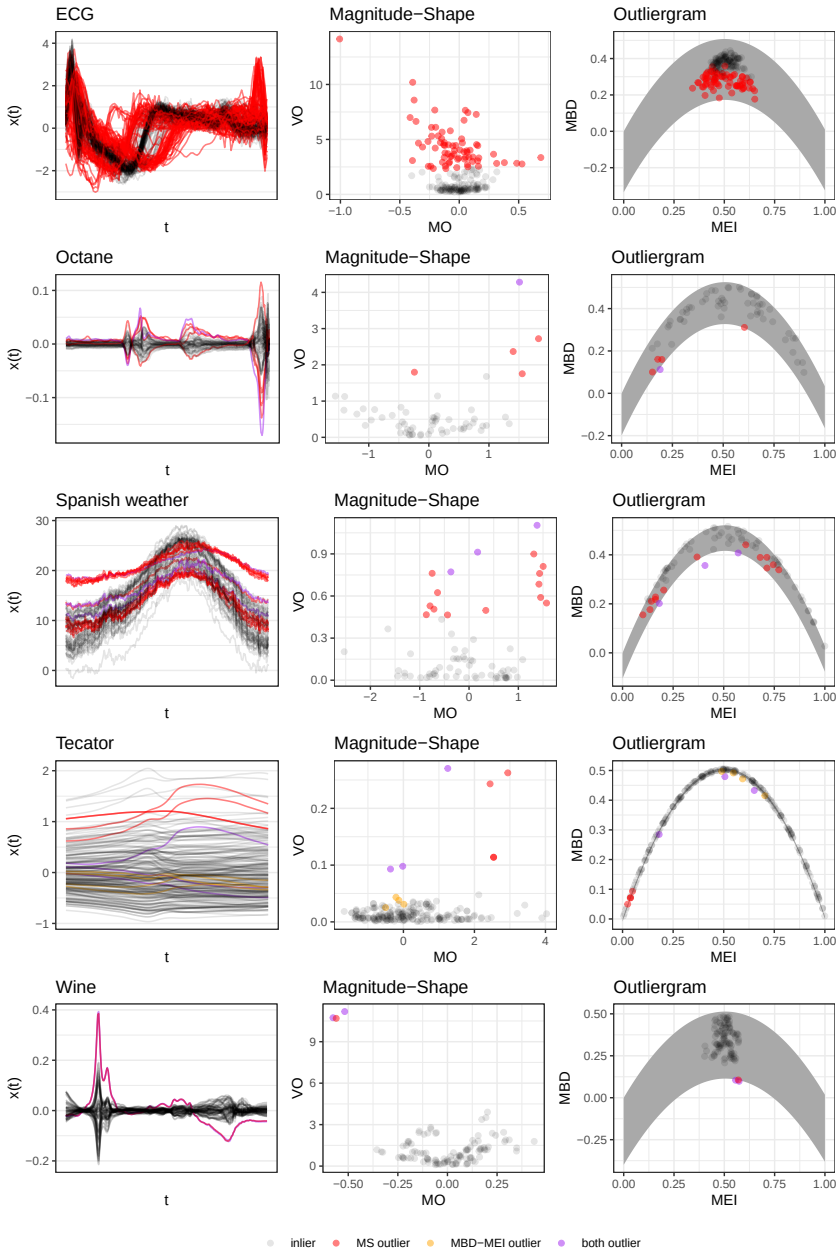

Figure A5. Left column: data; middle column: magnitude–shape plots of mean directional outlyingness (MO) versus variability of directional outlyingness (VO); right column: outliergram of the modified epigraph index (MEI) versus modified band depth (MBD) with the inlier region in grey. Curves and points are colored according to outlier status as diagnosed by `fdaoutlier::msplot` and/or `roahd::outliergram`.

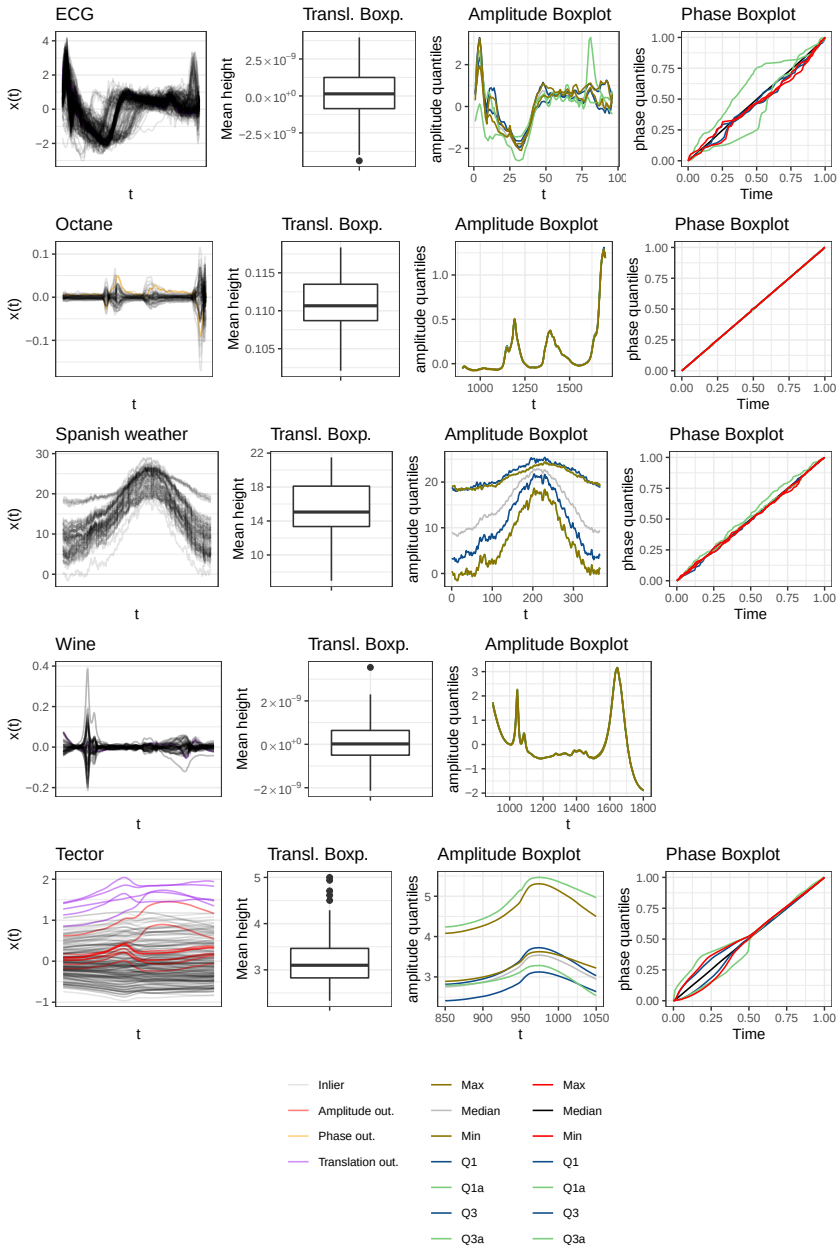

Figure A6. First column: data; second column: translation boxplots of average curve heights; third and fourth column: amplitude, respectively phase boxplots with the maximum and minimum extreme curves (Max, Min), the first and third quartile curves (Q1 and Q3), and the 0.05- and 0.95-quantile curves (Q1a, Q3a). Curves in the first column colored according to the outlier status by translational outlyingness, amplitude outlyingness, and phase outlyingness (the latter two as diagnosed by `fdasrvf::AmplitudeBoxplot` and `fdasrvf::AmplitudeBoxplot`). Note, for the Wine data, it was not possible to compute the phase boxplot.

Stats **2021**, *4* 999

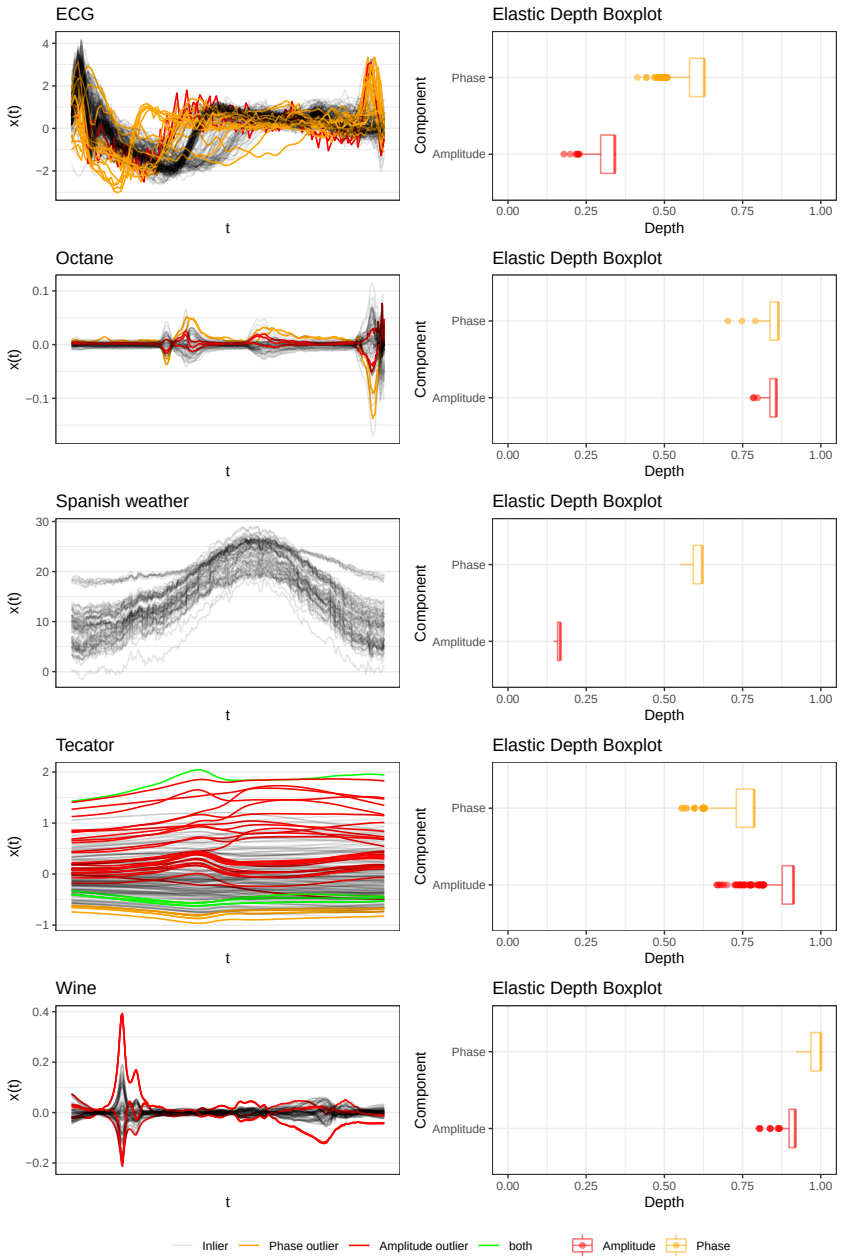

Figure A7. Left column: data; right column: elastic depth boxplots for amplitude and phase variability. Curves in the left column colored according to the outlier status by amplitude outlyingness and phase outlyingness as diagnosed by `elasticdepth::elastic_outliers`.

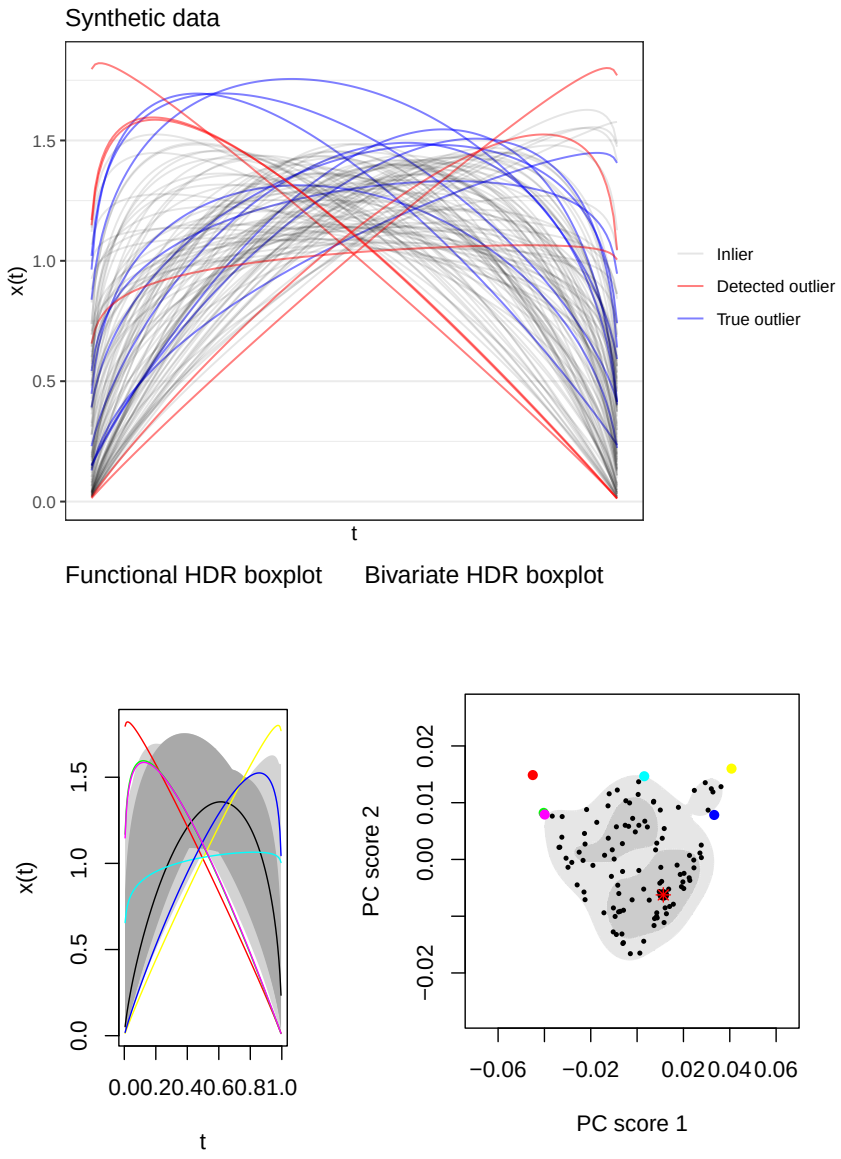

Figure A8. Upper row: synthetic data. Lower row, left column: functional HDR boxplot; right column: bivariate HDR boxplot. Colored curves/points are outliers according to a coverage probability of 0.05 for the functional HDR boxplot. HDR boxplots computed with `rainbow::fboxplot`.

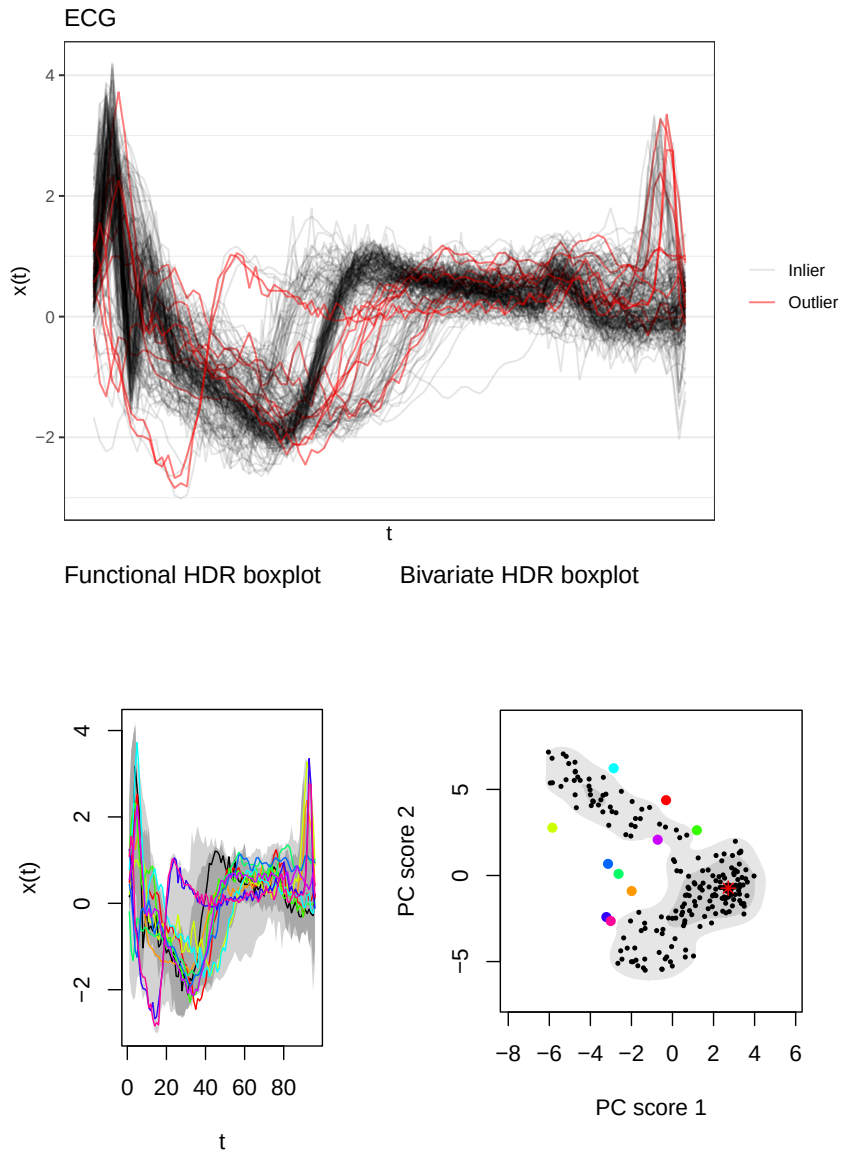

Figure A9. Upper row: ECG data. Lower row, left column: functional HDR boxplot; right column: bivariate HDR boxplot. Colored curves/points are outliers according to a coverage probability of 0.05 for the functional HDR boxplot. HDR boxplots computed with `rainbow::fboxplot`.

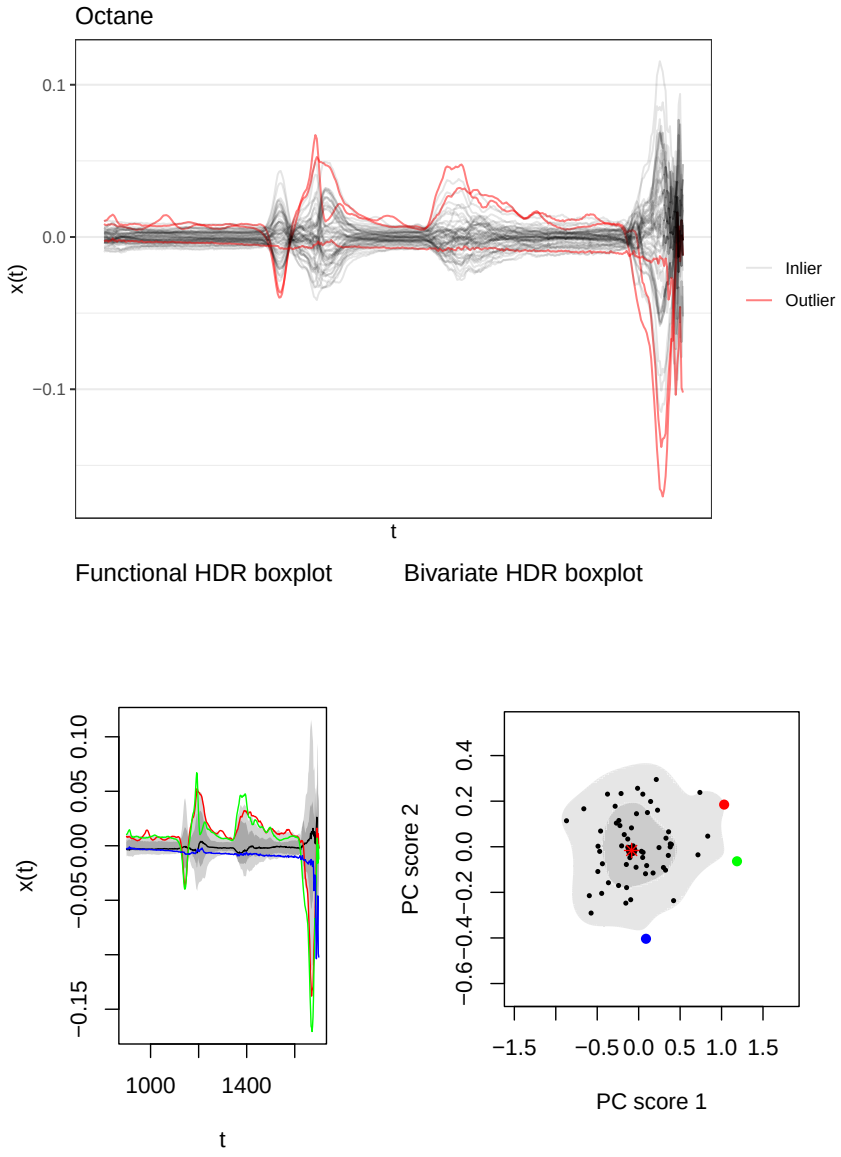

Figure A10. Upper row: Octane data. Lower row, left column: functional HDR boxplot; right column: bivariate HDR boxplot. Colored curves/points are outliers according to a coverage probability of 0.05 for the functional HDR boxplot. HDR boxplots computed with `rainbow::fboxplot`.

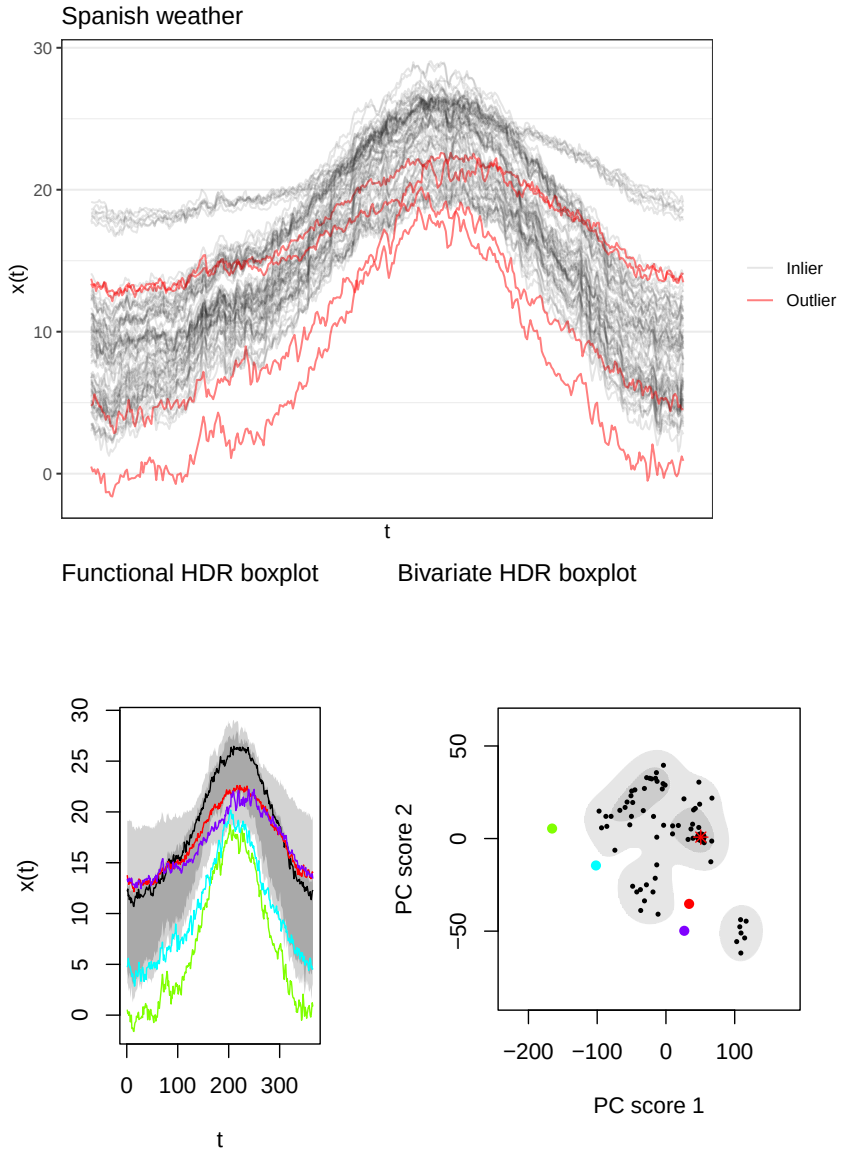

Figure A11. Upper row: Spanish weather data. Lower row, left column: functional HDR boxplot; right column: bivariate HDR boxplot. Colored curves/points are outliers according to a coverage probability of 0.05 for the functional HDR boxplot. HDR boxplots computed with `rainbow::fboxplot`.

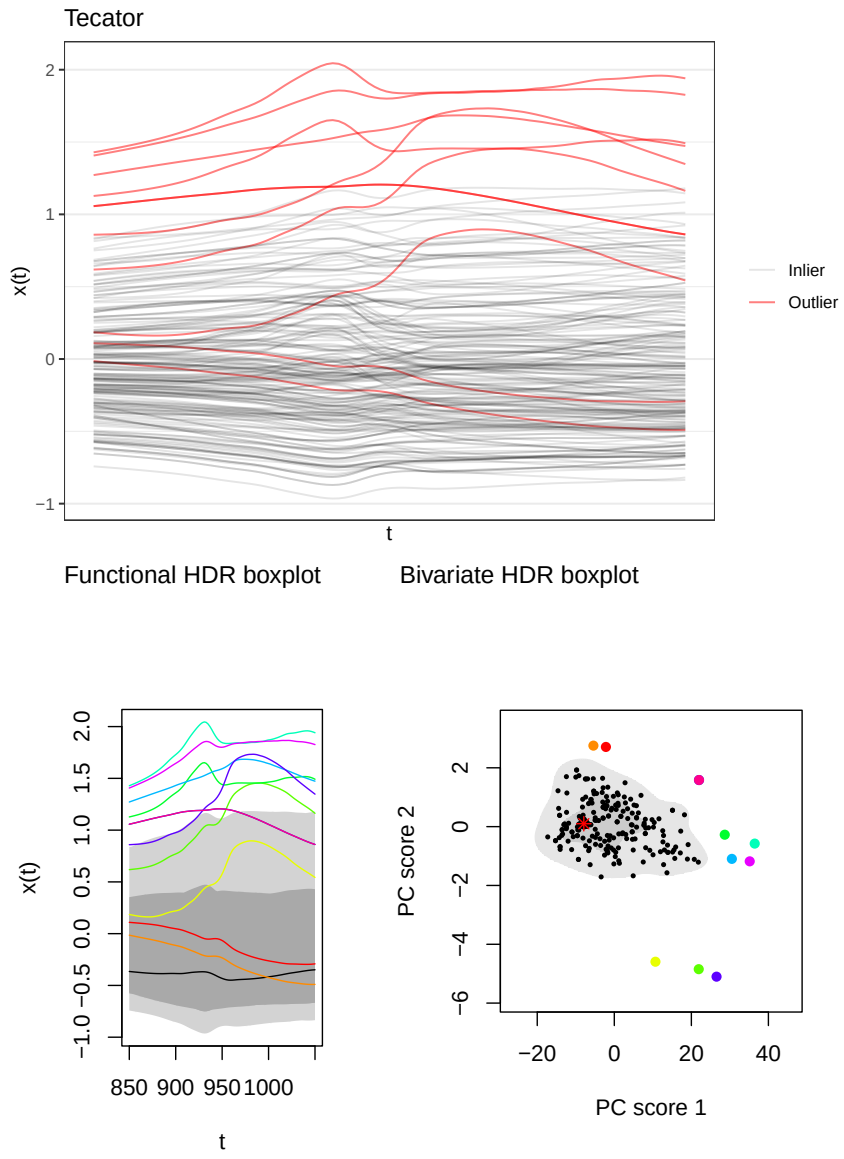

Figure A12. Upper row: Tecator data. Lower row, left column: functional HDR boxplot; right column: bivariate HDR boxplot. Colored curves/points are outliers according to a coverage probability of 0.05 for the functional HDR boxplot. HDR boxplots computed with `rainbow::fboxplot`.

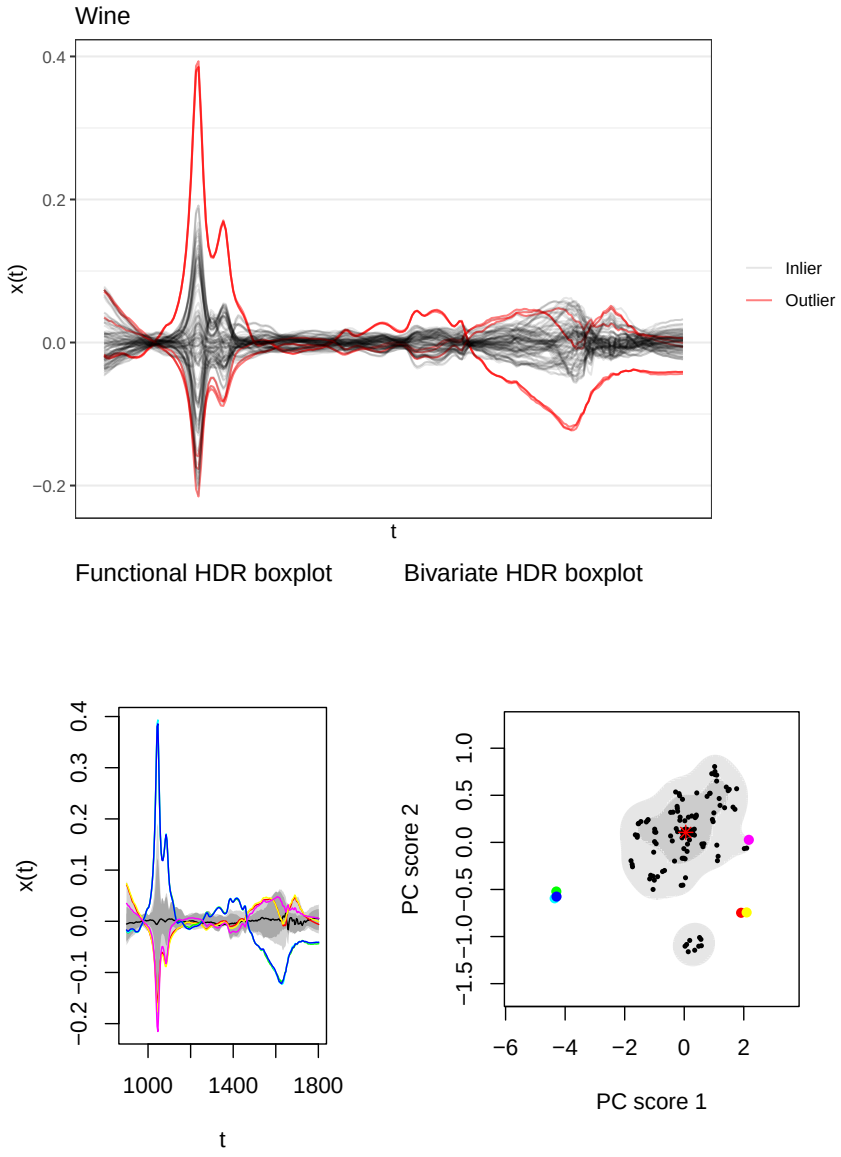

Figure A13. Upper row: Wine data. Lower row, left column: functional HDR boxplot; right column: bivariate HDR boxplot. Colored curves/points are outliers according a coverage probability of 0.05 for the functional HDR boxplot. HDR boxplots computed with `rainbow::fboxplot`.

Appendix E. In-Depth Analysis of Simulation Model 7

The analysis of the ECG data in Section 3.1 showed that embeddings can reveal much more (outlier) structure than can be represented by scores and labels. To illustrate the effects described in Appendix C, we conducted a similar qualitative analysis for an example dataset with observations sampled from Simulation Model 7; see Figure A14. The dataset consisted of 100 observations with 10 off-manifold or—in more informal terms: "true"—outliers. The functions were evaluated on 50 grid points. The analysis showed that a quantitative performance assessment alone may yield misleading results and again emphasizes the practical value of the geometric perspective and low-dimensional embeddings.

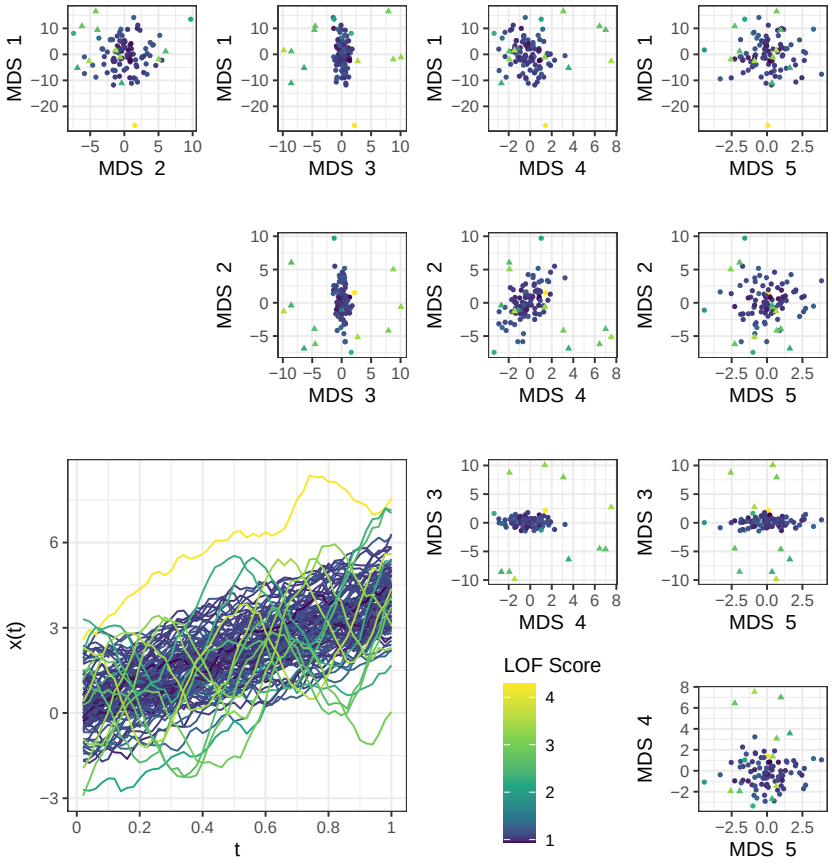

Figure A14. Model 7 data: scatterplot matrix of all 5 MDS embedding dimensions and curves; lighter colors for higher LOF score of 5D embeddings. True outliers depicted as triangles. Note that the true outliers are clearly separated from the rest of the data in embedding subspace 3 vs. 4.

First of all, note that the AUC computed for this specific dataset was 0.9, thus close to the median AUC value for LOF applied to MDS embeddings of Model 7 data, as depicted in Figure A3. Nevertheless, the "true outliers" are clearly separable in a 5D MDS embedding. As Figure A14 shows, they are clearly separable in the subspace spanned by the third and fourth embedding dimension. Note, moreover, that there is an outlying observation with an extreme shift, which also obtains a high LOF score. This observation is not labeled as a "true outlier", as it stems from $\mathcal{M}_c$. This example shows that evaluation approaches

for outlier detection methods that are based on "true outliers" may not always reflect the outlier structure adequately and may result in misleading conclusions. However, those approaches are frequently used to compare and assess different outlier-detection methods. Again, this illustrates the additional value low-dimensional embeddings have for outlier detection as such aspects become accessible.

Finally, note that the DO/MS-plots are not sensitive to vertical shift outliers as the extreme shift outlier is neither scored high based on DO nor labeled as an outlier based on the MS-plot; see Figure A15.

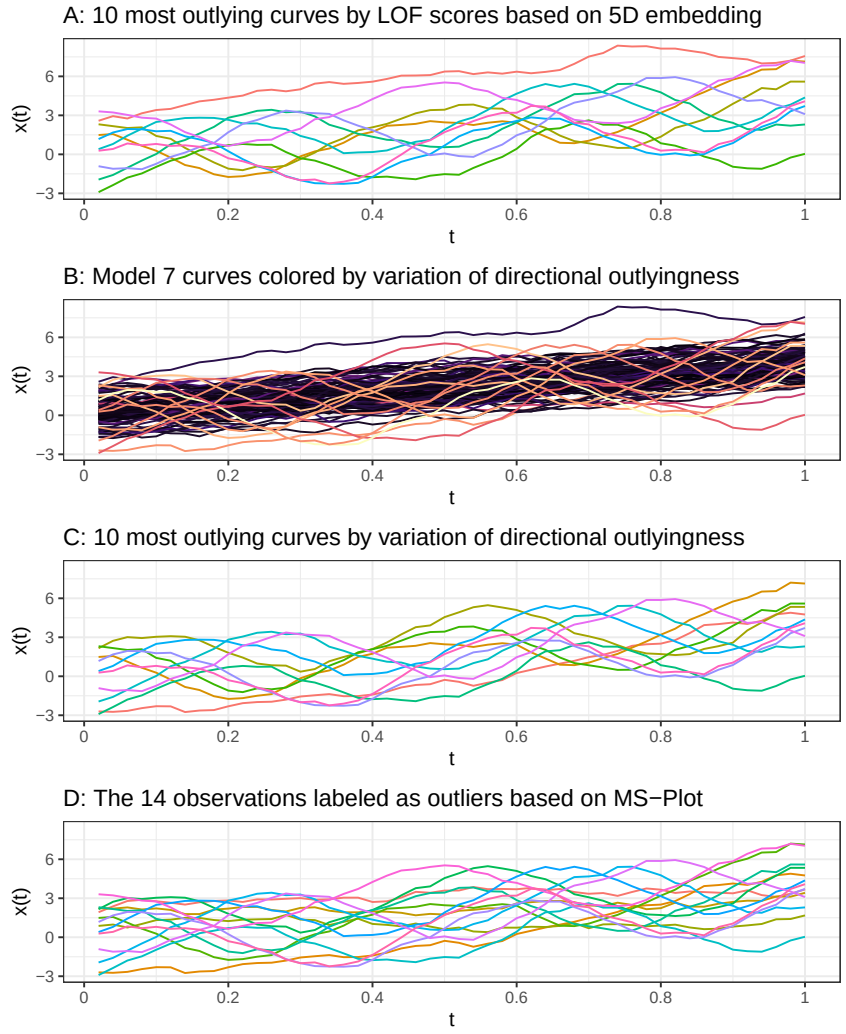

Figure A15. Model 7 data: the LOF on MDS embeddings in contrast to directional outlyingness.

Appendix F. Examples of the DGPs Used for the Quantitative Evaluation

Depicted in Figure A16 are two example datasets for each of the data-generating processes (DGPs) used in Section 3.2 for the comparison of the different outlier-detection methods.

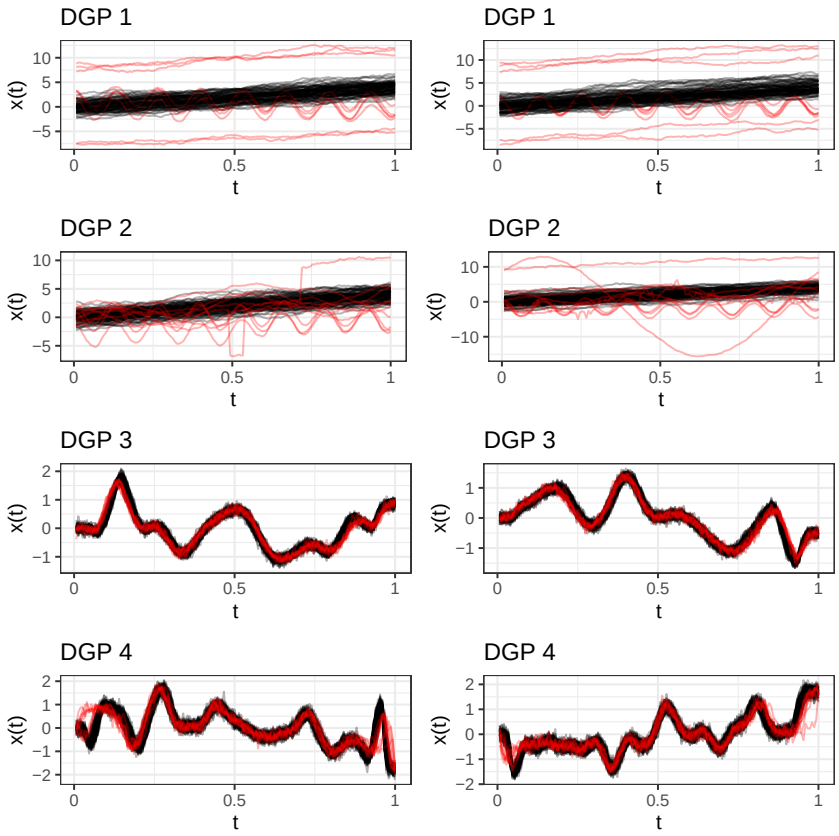

Figure A16. Example datasets for the DGPs used in the simulation study (2 each). Inliers in black; outliers in red. Outlier ratio 0.1; $n = 100$.

Appendix G. ArrowHead Data

Depicted in Figure A17 are the ArrowHead data used in Section 3.3.

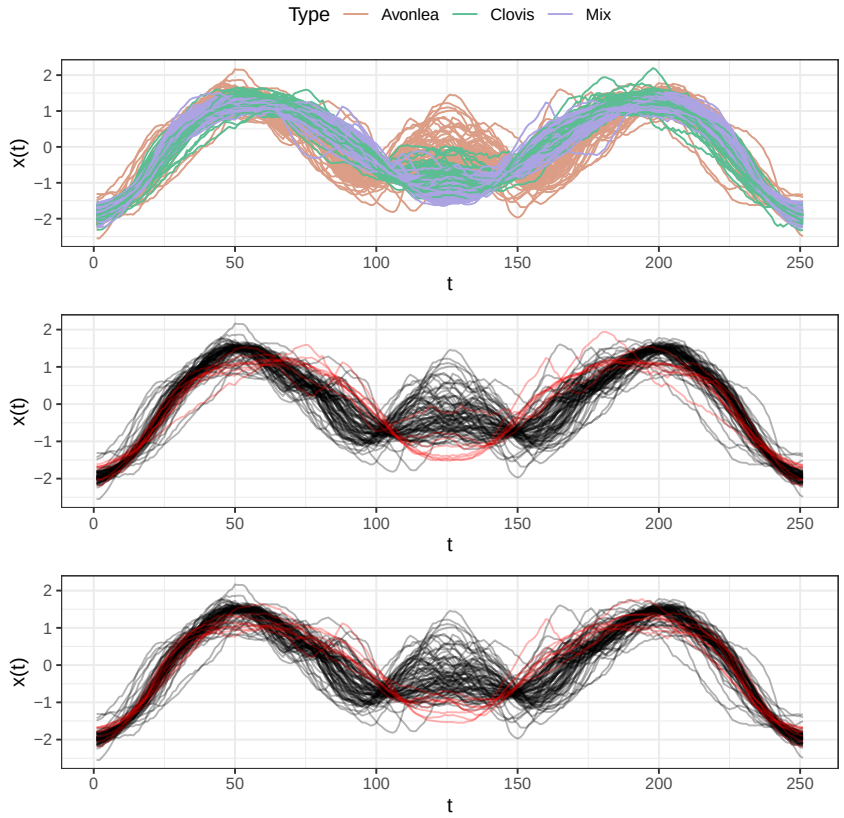

Figure A17. ArrowHead data. Top: the complete dataset. Middle and bottom: two example outlier datasets. Inliers from class "Avonlea" in black; outliers sampled from classes "Clovis" and "Mix" in red. Outlier ratio 0.1.

References

1. Dai, W.; Mrkvička, T.; Sun, Y.; Genton, M.G. Functional outlier detection and taxonomy by sequential transformations. *Comput. Stat. Data Anal.* **2020**, *149*, 106960. [CrossRef]
2. Arribas-Gil, A.; Romo, J. Discussion of "Multivariate functional outlier detection". *Stat. Methods Appl.* **2015**, *24*, 263–267. [CrossRef]
3. Hubert, M.; Rousseeuw, P.J.; Segaert, P. Multivariate functional outlier detection. *Stat. Methods Appl.* **2015**, *24*, 177–202. [CrossRef]
4. Ma, Y.; Fu, Y. *Manifold Learning Theory and Applications*; CRC Press: Boca Raton, FL, USA, 2012.
5. Lee, J.A.; Verleysen, M. *Nonlinear Dimensionality Reduction*; Springer Science & Business Media: New York, NY, USA, 2007.
6. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: Identifying density-based local outliers. In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, Dallas, TX, USA, 15–18 May 2000; pp. 93–104.
7. Ramsay, J.O.; Silverman, B.W. *Functional Data Analysis*, 2nd ed.; Springer Series in Statistics; Springer: New York, NY, USA, 2005.
8. Hernández, N.; Muñoz, A. Kernel Depth Measures for Functional Data with Application to Outlier Detection. In *Artificial Neural Networks and Machine Learning–ICANN 2016*; Villa, A.E., Masulli, P., Pons Rivero, A.J., Eds.; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2016; pp. 235–242.
9. Harris, T.; Tucker, J.D.; Li, B.; Shand, L. Elastic depths for detecting shape anomalies in functional data. *Technometrics* **2021**, *63*, 466–476. [CrossRef]
10. Sawant, P.; Billor, N.; Shin, H. Functional outlier detection with robust functional principal component analysis. *Comput. Stat.* **2012**, *27*, 83–102. [CrossRef]

11. Staerman, G.; Mozharovskyi, P.; Clémençon, S.; d'Alché Buc, F. Functional isolation forest. In Proceedings of the Eleventh Asian Conference on Machine Learning, Nagoya, Japan, 17–19 November 2019; Lee, W.S., Suzuki, T., Eds.; Volume 10, pp. 332–347.
12. Vinue, G.; Epifanio, I. Robust archetypoids for anomaly detection in big functional data. *Adv. Data Anal. Classif.* **2021**, *15*, 437–462. [CrossRef]
13. Rousseeuw, P.J.; Raymaekers, J.; Hubert, M. A measure of directional outlyingness with applications to image data and video. *J. Comput. Graph. Stat.* **2018**, *27*, 345–359. [CrossRef]
14. Dai, W.; Genton, M.G. Directional outlyingness for multivariate functional data. *Comput. Stat. Data Anal.* **2019**, *131*, 50–65. [CrossRef]
15. Xie, W.; Kurtek, S.; Bharath, K.; Sun, Y. A Geometric Approach to Visualization of Variability in Functional data. *J. Am. Stat. Assoc.* **2017**, *112*, 979–993. [CrossRef]
16. Hyndman, R.J.; Shang, H.L. Rainbow plots, bagplots, and boxplots for functional data. *J. Comput. Graph. Stat.* **2010**, *19*, 29–45. [CrossRef]
17. Ali, M.; Jones, M.W.; Xie, X.; Williams, M. TimeCluster: dimension reduction applied to temporal data for visual analytics. *Vis. Comput.* **2019**, *35*, 1013–1026. [CrossRef]
18. Yu, G.; Zou, C.; Wang, Z. Outlier Detection in Functional Observations with Applications to Profile Monitoring. *Technometrics* **2012**, *54*, 308–318. [CrossRef]
19. Chen, D.; Müller, H.G. Nonlinear manifold representations for functional data. *Ann. Stat.* **2012**, *40*, 1–29. [CrossRef]
20. Dimeglio, C.; Gallón, S.; Loubes, J.M.; Maza, E. A robust algorithm for template curve estimation based on manifold embedding. *Comput. Stat. Data Anal.* **2014**, *70*, 373–386. [CrossRef]
21. Herrmann, M.; Scheipl, F. Unsupervised Functional Data Analysis via Nonlinear Dimension Reduction. *arXiv* **2020**, arXiv:2012.11987.
22. Cuevas, A. A partial overview of the theory of statistics with functional data. *J. Stat. Plan. Inference* **2014**, *147*, 1–23. [CrossRef]
23. Malkowsky, E.; Rakočević, V. *Advanced Functional Analysis*; CRC Press: Boca Raton, FL, USA, 2019.
24. Polonik, W. Minimum volume sets and generalized quantile processes. *Stoch. Process. Their Appl.* **1997**, *69*, 1–24. [CrossRef]
25. Ojo, O.; Lillo, R.E.; Anta, A.F. Outlier Detection for Functional Data with R Package fdaoutlier. *arXiv* **2021**, arXiv:2105.05213.
26. Zimek, A.; Filzmoser, P. There and back again: Outlier detection between statistical reasoning and data mining algorithms. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2018**, *8*, e1280. [CrossRef]
27. Cox, M.A.; Cox, T.F. Multidimensional scaling. In *Handbook of Data Visualization*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 315–347.
28. Tenenbaum, J.B.; Silva, V.D.; Langford, J.C. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science* **2000**, *290*, 2319–2323. [CrossRef]
29. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.
30. McInnes, L.; Healy, J.; Melville, J. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv* **2020**, arXiv:1802.03426.
31. Gangbo, W.; Li, W.; Osher, S.; Puthawala, M. Unnormalized optimal transport. *J. Comput. Phys.* **2019**, *399*, 108940. [CrossRef]
32. Bagnall, A.; Lines, J.; Bostrom, A.; Large, J.; Keogh, E. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Min. Knowl. Discov.* **2017**, *31*, 606–660. [CrossRef]
33. Olszewski, R.T. Generalized Feature Extraction for Structural Pattern Recognition in Time-Series Data. Ph.D. Thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2001.
34. Dai, W.; Genton, M.G. Multivariate functional data visualization and outlier detection. *J. Comput. Graph. Stat.* **2018**, *27*, 923–934. [CrossRef]
35. Shang, H.L.; Hyndman, R.J. *fds: Functional Data Sets*; R Package Version 1.8; R package; 2018.
36. Kalivas, J.H. Two datasets of near infrared spectra. *Chemom. Intell. Lab. Syst.* **1997**, *37*, 255–259. [CrossRef]
37. Febrero-Bande, M.; Oviedo de la Fuente, M. Statistical Computing in Functional Data Analysis: The R Package fda.usc. *J. Stat. Softw.* **2012**, *51*, 1–28. [CrossRef]
38. Ferraty, F.; Vieu, P. *Nonparametric Functional Data Analysis: Theory and Practice*; Springer Science & Business Media: New York, NY, USA, 2006.
39. Holland, J.; Kemsley, E.; Wilson, R. Use of Fourier transform infrared spectroscopy and partial least squares regression for the detection of adulteration of strawberry purees. *J. Sci. Food Agric.* **1998**, *76*, 263–269. [CrossRef]
40. Mead, A. Review of the development of multidimensional scaling methods. *J. R. Stat. Soc. Ser.* **1992**, *41*, 27–39. [CrossRef]
41. Arribas-Gil, A.; Romo, J. Shape outlier detection and visualization for functional data: the outliergram. *Biostatistics* **2014**, *15*, 603–619. [CrossRef]
42. Ieva, F.; Paganoni, A.M.; Romo, J.; Tarabelloni, N. `roahd` Package: Robust Analysis of High Dimensional Data. *R J.* **2019**, *11*, 291–307. [CrossRef]
43. Shang, H.L.; Hyndman, R. *Rainbow: Bagplots, Boxplots and Rainbow Plots for Functional Data*; R package version 3.6; R package; 2019.
44. Huang, H.; Sun, Y. A decomposition of total variation depth for understanding functional outliers. *Technometrics* **2019**, *61*, 445–458. [CrossRef]

45. Ojo, O.T.; Lillo, R.E.; Fernandez Anta, A. *fdaoutlier: Outlier Detection Tools for Functional Data Analysis*; R package version 0.2.0.; R package; 2021.
46. Tucker, J.D. *fdasrvf: Elastic Functional Data Analysis*; R package version 1.9.7.; R package; 2021.
47. Dau, H.A.; Bagnall, A.; Kamgar, K.; Yeh, C.C.M.; Zhu, Y.; Gharghabi, S.; Ratanamahatana, C.A.; Keogh, E. The UCR time series archive. *IEEE/CAA J. Autom. Sin.* **2019**, *6*, 1293–1305. [CrossRef]
48. Ye, L.; Keogh, E. Time series shapelets: A new primitive for data mining. In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, 28 June–1 July 2009; pp. 947–956.
49. Rakthanmanon, T.; Campana, B.; Mueen, A.; Batista, G.; Westover, B.; Zhu, Q.; Zakaria, J.; Keogh, E. Searching and mining trillions of time series subsequences under dynamic time warping. In Proceedings of the 18th ACM SIGKDD international Conference on Knowledge Discovery and Data Mining, Beijing, China, 12–16 August 2012; pp. 262–270.
50. Lemire, D. Faster retrieval with a two-pass dynamic-time-warping lower bound. *Pattern Recognit.* **2009**, *42*, 2169–2180. [CrossRef]
51. Fuchs, K.; Gertheiss, J.; Tutz, G. Nearest neighbor ensembles for functional data with interpretable feature selection. *Chemom. Intell. Lab. Syst.* **2015**, *146*, 186–197. [CrossRef]
52. Narayan, A.; Berger, B.; Cho, H. Assessing single-cell transcriptomic variability through density-preserving data visualization. *Nat. Biotechnol.* **2021**, *39*, 765–774. [CrossRef] [PubMed]
53. De Silva, V.; Tenenbaum, J.B. Global versus local methods in nonlinear dimensionality reduction. *NIPS* **2002**, *15*, 705–712.
54. Brandes, U.; Pich, C. Eigensolver methods for progressive multidimensional scaling of large data. In *International Symposium on Graph Drawing*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 42–53.
55. Ingram, S.; Munzner, T.; Olano, M. Glimmer: Multilevel MDS on the GPU. *IEEE Trans. Vis. Comput. Graph.* **2008**, *15*, 249–261. [CrossRef] [PubMed]
56. Clémençon, S.; Thomas, A. Mass volume curves and anomaly ranking. *Electron. J. Stat.* **2018**, *12*, 2806–2872. [CrossRef]

9. A Geometric Framework for Outlier Detection in High-Dimensional Data

Chapter 9 generalizes the insights gained in the study presented in Chapter 8. It is demonstrated that the approach straightforwardly extends to other non-tabular, high-dimensional data types such as graphs and images practically and theoretically. Moreover, it provides a review of conceptual aspects of outlier detection in general discussed in the literature. In particular, there are two vague notions of outliers that are not sufficiently reflected by existing conceptualizations. These notions can be made precise with the proposed geometrical concepts differentiating off-manifold and on-manifold outliers, which are termed structural and distributional outliers in this contribution.

Contributing article:

Herrmann, M., Pfisterer, F., & Scheipl, F. (2022). A geometric framework for outlier detection in high-dimensional data. arXiv preprint arXiv:2207.00367. https://arxiv.org/abs/2207.00367

Author contributions:

Moritz Herrmann had the idea of dealing with the topic in this way and wrote the paper. Florian Pfisterer and Fabian Scheipl made contributions by continuously revising the manuscript and adding ideas.

Supplementary material available at:

Code and data: https://github.com/HerrMo/geo-outlier-framework

A geometric framework for outlier detection in high-dimensional data

Moritz Herrmann*, Florian Pfisterer, and Fabian Scheipl

Department of Statistics, Ludwig Maximilians University, Munich, Germany

Abstract

Outlier or anomaly detection is an important task in data analysis. We discuss the problem from a geometrical perspective and provide a framework which exploits the metric structure of a data set. Our approach rests on the *manifold assumption*, i.e., that the observed, nominally high-dimensional data lie on a much lower dimensional manifold and that this intrinsic structure can be inferred with manifold learning methods. We show that exploiting this structure significantly improves the detection of outlying observations in high dimensional data. We also suggest a novel, mathematically precise and widely applicable distinction between *distributional* and *structural* outliers based on the geometry and topology of of the data manifold that clarifies conceptual ambiguities prevalent throughout the literature. Our experiments focus on functional data as one class of structured high-dimensional data, but the framework we propose is completely general and we include image and graph data applications. Our results show that the outlier structure of high-dimensional and non-tabular data can be detected and visualized using manifold learning methods and quantified using standard outlier scoring methods applied to the manifold embedding vectors.

1 Introduction

Detecting atypical observations that deviate substantially from the bulk of the data is an important task in data analysis with applications across domains like, e.g., intrusion detection (Zhang & Zulkernine, 2006), medical imaging (Fritsch et al., 2012), or network analysis (Azcorra et al., 2018). The most common terms for this task are *outlier* or *anomaly detection*, but many different terms are used (Zimek & Filzmoser, 2018). Although there is a vast amount of literature on the topic, there is neither a commonly accepted, precise definition of what exactly constitutes outliers or anomalies, nor agreement on whether these two terms are synonymous. As Unwin (2019, p. 635) puts it:

> "Outliers are a complicated business. It is difficult to define what they are, it is difficult to identify them, and it is difficult to assess how they affect analyses."

Overviews on the topic are given by Zimek et al. (2012) or Goldstein & Uchida (2016) from a computer science perspective, and by Rousseeuw & Leroy (2005) or Unwin (2019) from a

*Corresponding author, e-mail: moritz.herrmann@stat.uni-muenchen.de, Department of Statistics, Ludwig Maximilians University Munich, Ludwigstr. 33, D-80539, Munich, Germany.

statistical perspective. Kandanaarachchi & Hyndman (2020) provide a short summary including both perspectives, while Campos et al. (2016) as well as Marques et al. (2020) focus on the evaluation of unsupervised outlier detection. Zimek & Filzmoser (2018) provide a comprehensive survey bringing together both perspectives with in-depth epistemological discussion. In particular, Zimek & Filzmoser (2018) discuss that there are two different notions of outliers and different terms used to describe these notions – including, for example, *apparent*, *discrepant*, *real*, *contaminating*, or *true* outlier – in the literature. From this discussion, it can be inferred that (1) Zimek & Filzmoser (2018, p. 7) distinguish "true" and "apparent" outliers and consider "those objects as '(true) outliers' that have been 'generated by a different mechanism' than the remainder or major part of the data or than the whatsoever defined reference set", (2) there is neither a clear understanding of how these two notions are different and actually manifest in practice nor (3) a "language" to precisely describe the problem theoretically.

In the more statistically flavored literature, the problem of unsupervised outlier detection is usually tackled by defining outliers based on a single probability distribution P. If P allows for a density, outliers are simply observations in low-density regions. From this perspective, we have *distributional outliers* whose outlyingness is defined relative to a single probability distribution. The notion of *distributional outliers* is easy to define precisely in probabilistic terms, for example, based on minimum level sets (Scott & Nowak, 2006) or M-estimation (Clémençon & Jakubowicz, 2013), and has yielded a multitude of results and algorithms. In practical terms, this requires access to (an estimate of) the underlying density and finding a suitable (local) density level below which observations are to be classified as outliers. Note that both are infeasible for general, non-tabular data types like shapes, functions, or images whose domains frequently do not admit probability densities. However, Zimek & Filzmoser (2018) emphasize that "observations which are in the extremes of the model distribution [i.e., distributional outlier] should be distinguished from 'real' outliers (contaminants)" (Zimek & Filzmoser, 2018, p. 13). This second notion of outliers ("true" or "real" outliers) is not reflected by the statistical concept because such outliers are assumed to be observations generated by a different data-generating process. This is reflected in statements like "different mechanism" or "any observation that is not a realization from the target distribution" (Beckman & Cook, 1983, p. 121). That means, for the second notion ("real outliers") it is implicitly assumed that outliers are not independent and identically distributed (IID) observations. So next to *distributional outliers* there are also *structural outliers* whose outlyingness is caused by the structural differences between the underlying data generating processes. The two outlier types are complementary and both are necessary to fully address the challenges of outlier detection. In contrast to *distributional outliers*, *structural outliers* are much more difficult to formalize, but also more general.

With this work, we intend to "broaden" the view on the problem of unsupervised outlier detection to account for the two notions of outliers present in the literature. We show that a *geometric* approach to the problem, which does not require the availability of probability densities defined over the data space but only some metric structure (i.e., suitable dissimilarity or distance measures), allows for a more precise conceptualization and terminology. To do so, we focus on building up intuition and demonstrating the application of these concepts to diverse and comprehensive practical examples and visualizations. However, to be able to "speak" of this new perspective without referring to vague and subjective perceptions as done previously (see Zimek & Filzmoser, 2018), we also consider it necessary to introduce a certain degree of mathematical terminology. The provided degree of formality is exhausted in the definition of *distributional* and *structural* outliers in precise mathematical terms and thus serves the need for precise terminology but does not overload the work with more formalism than we think necessary to contribute to the overall scope of the study. Readers interested in more rigorous mathematical approaches to

infer structures in data may, for example, consult Mordohai & Medioni (2010) for dimensionality estimation and manifold learning based on tensor voting, Niyogi et al. (2011) for a topological perspective on unsupervised learning, Guan & Loew (2021) of a distance-based measure of class separability, and Kandanaarachchi & Hyndman (2020) for outlier detection in tabular data based on dimensionality reduction.

The rest of the paper is structured as follows. Section 2 describes the scope and contribution of the study and outlines its background and related work. The proposed theoretical framework is defined in section 3 and its practical relevance is demonstrated in section 4 using qualitative and quantitative experiments for a variety of data sets of different data types. Section 5 discusses our findings and the resulting conceptual implications, before we conclude in section 6.

2 Preliminaries

2.1 Scope and contribution of the study

With this focus article, we intend to draw connections between different conceptual aspects provided in the overview article by Zimek & Filzmoser (2018) and as proposed by us in a paper focusing on functional data analysis (FDA). Therefore, we recapitulate the underlying conceptualization presented in the earlier paper in a more general form and different terminology in Section 3. This framework builds on principles from *manifold learning* (Lee & Verleysen, 2007; Ma & Fu, 2011), i.e., dimension reduction methods that infer the intrinsic lower-dimensional manifold structure of high-dimensional data and yield low-dimensional vector representations of the data. This perspective allows us to formalize structural and distributional outliers jointly in a single mathematical framework, where structural outliers are data that are separate from the main data manifold, and distributional outliers are data that are situated at the periphery of, but still on the main data manifold. While the first paper exclusively focused on the functional data setting, the present focus article generalizes the underlying conceptualization of outlier detection to other data types. This is straightforward theoretically but has important general conceptual implications that have never been described in detail and demonstrated on diverse real data problems before. In particular, we draw connections between and provide a unifying perspective on different data types (functions, images, graphs, tabular data) which were previously often treated separately from a theoretical as well as a practical perspective, in particular when it comes to outlier detection.

The main contribution of this review paper is to discuss and demonstrate two conceptual aspects of outlier detection in general. First of all, as already outlined, there seems to be a lack of clarity about what defines outliers, evidenced also by the plethora of terms used to describe the issue (Zimek & Filzmoser, 2018). Several recent reviews on the topic also point out this conceptual ambiguity (Goldstein & Uchida, 2016; Unwin, 2019; Zimek & Filzmoser, 2018). In particular, the comprehensive overview of Zimek and Filzmoser (2018, p. 4) devotes a complete section to the question of "what an 'outlier' possibly means". Recall that they define "true outliers" as objects "that have been 'generated by a different mechanism' than the remainder or major part of the data or than the whatsoever defined reference set" and distinguish them from "objects that appear to be outliers (independent of whether or not they actually are (true) outliers") (Zimek & Filzmoser, 2018, p. 7). As we will show, the geometrical framework provides suitable mathematical terminology to delineate "true" and "apparent" outliers much more cleanly and thus reduces the conceptual ambiguity that surrounds the topic: We transfer concepts established in manifold learning to the problem of outlier detection, deriving a novel underlying

conceptualization of the problem of outlier detection that (1) is capable of reflecting two types of outliers, (2) replaces vague notions of "real", "contaminant", or "apparent" outliers with a precise definition, (3) incorporates the well-established concept of *distributional* outliers in a unified fashion. With this, we can abandon vague notions of outlier subtypes in favor of two precisely defined concepts.
Second, our framework also suggests that outlier detection in high-dimensional (and/or non-tabular) data is not necessarily more challenging than in low-dimensional settings once the underlying manifold structure is recovered and exploited. This is important because high dimensionality is often reported to be particularly problematic for outlier detection and many outlier detection methods break down or at least face particular challenges in such settings (Aggarwal, 2017; Aggarwal & Yu, 2001; Goldstein & Uchida, 2016; Kamalov & Leung, 2020; Navarro-Esteban & Cuesta-Albertos, 2021; Ro et al., 2015; Thudumu et al., 2020; Xu et al., 2018; Zimek et al., 2012, e.g.).
To highlight these aspects, we again provide simple and easily accessible functional data examples to demonstrate the principal practical implications (in addition to the recapitulation of the theoretical conceptualization). Functional data analysis (Ramsay & Silverman, 2005, e.g.) deals with data that are (discretized) realizations of stochastic processes over a compact domain. Functional data is well suited to illustrate the underlying conceptualization both practically and theoretically because it is usually highly structured (the manifold assumption is specifically realistic and useful), theoretically/analytically well accessible, and easily visualized in bulk. Beyond the FDA setting, we also use examples of other data types including image, graph, curve, and tabular data.
Finally, our framework is fully general and does not rely on a specific combination of manifold learning and outlier detection methods. To demonstrate its practical performance, we show that one of the simplest and most established manifold learning methods – Multidimensional Scaling (MDS) (Cox & Cox, 2008) – combined with a standard outlier detection algorithm – Local Outlier Factors (LOF) (Breunig et al., 2000) – already yields a flexible, reliable, and generally applicable recipe for outlier detection and visualization in complex, high-dimensional data.

2.2 Background and related work

The fundamental assumption of manifold learning is that the high-dimensional data observed in a D-dimensional space $\mathcal{H}$ actually lie on a d-dimensional manifold $\mathcal{M} \subset \mathcal{H}$ with $d < D$. Manifold learning methods yield an *embedding* function $e : \mathcal{H} \to \mathcal{Y}$ from the high-dimensional data space to a low-dimensional embedding space $\mathcal{Y}$ such that the configuration of embedded data reflects the characteristics of $\mathcal{M}$. The terms manifold learning and nonlinear dimension reduction are often used interchangeably (Lee & Verleysen, 2007; Ma & Fu, 2011). Typically, the fundamental step is to compute distances between the high-dimensional observations. Methods based on this approach are, for example, Multidimensional Scaling (MDS) (Cox & Cox, 2008), Isomap (Tenenbaum et al., 2000), diffusion maps (Coifman & Lafon, 2006), local linear embeddings (Roweis & Saul, 2000), Laplacian eigenmaps (Belkin & Niyogi, 2003), t-distributed stochastic neighborhood embeddings (t-SNE) (Maaten & Hinton, 2008), and uniform manifold approximation and projection (UMAP) (McInnes et al., 2018), to name only a few. The methods differ in how they infer the manifold structure from these distances and how they obtain low-dimensional embedding vectors from these.
Despite their promising results in other settings, manifold learning methods have not found application for outlier detection to a significant extent so far. Kandanaarachchi & Hyndman (2020) define an outlier detection method explicitly based on dimension reduction, while Pang et al. (2018) make use of ranking model-based representation learning. However, they do not

provide a general conceptual framework and focus on tabular data. For functional data, Xie et al. (2017) introduce a geometric approach that decomposes functional observations into amplitude, phase, and shift components in order to identify specific types of outliers. However, the approach is only applicable to functional data and does not make use of the intrinsic structure of the functional observations from a manifold learning perspective. Ali et al. (2019) analyze time series data using $2\mathbf{D}$-embeddings obtained from manifold methods for outlier detection and clustering and Toivola et al. (2010) compare specific dimensionality reduction techniques for outlier detection in structural health monitoring, but both focus on practical considerations and do not provide a theoretical underpinning. Another line of work focuses on projection-based outlier detection, for example for high-dimensional Gaussian data (Navarro-Esteban & Cuesta-Albertos, 2021), financial time series (Loperfido, 2020), or functional data (Ren et al., 2017).

3 Geometrical framework for outlier detection

The framework we propose generalizes an approach for outlier detection in functional data developed recently (Herrmann & Scheipl, 2021). Since the approach exploits the metric structure of a functional data set, it is straightforward to generalize it to other data types, both from a theoretical as well as a practical perspective. Theoretically, the observation space needs to be a metric space, i.e. it needs to be equipped with a metric. Practically, there only needs to be a suitable distance measure to compute pairwise distances between observations. Two assumptions are fundamental for the framework. First of all, the *manifold assumption* that observed high-dimensional data lie on or close to a (low-dimensional) manifold. Note that functional data typically contain a lot of structure, and it is often reasonable to assume that only a few modes of variation suffice to describe most of the information contained in the data, i.e., such functional data often have low intrinsic dimension, at least approximately, see Figure 1 for a simple synthetic example. Similar remarks hold for other data types such as image data (Lee & Verleysen, 2007; Ma & Fu, 2011). Secondly, it is assumed that outliers are either *structural outliers* – or in the terminology of Zimek and Filzmoser (2018, p. 10) "real outliers" stemming from a different data generating process than the bulk of the data – or *distributional* outliers, observations that are structurally similar to the main data but still appear outlying in some sense. We make these notions mathematically precise in the remainder of this section based on the exposition in Herrmann & Scheipl (2021) before we demonstrate the practical relevance of the framework in section 4 and summarize its general conceptual implications in section 5.

Given a high-dimensional observation space $\mathcal{H}$ of dimension D, a d-dimensional parameter space $\Theta \subset \mathbb{R}^d$, such that the elements $\theta_i \in \Theta$ are realizations of the probability distribution P over the domain $\mathbb{R}^d$, i.e., $\theta_i \sim P$, and given an embedding space $\mathcal{Y} \subset \mathbb{R}^{d'}$, define the mappings ϕ and e so that

$$\Theta \xrightarrow{\phi} \mathcal{M}_{\mathcal{H}} \xrightarrow{e} \mathcal{Y},$$

with $\mathcal{M}_{\mathcal{H}} \subset \mathcal{H}$ a manifold in the observation space. The structure of $\mathcal{M}_{\mathcal{H}}$ is determined by the structure and dimensionality of Θ, P, and the map ϕ, which is isometric for the appropriate metrics in Θ and $\mathcal{H}$. Conceptually, the low-dimensional parameter space Θ represents the modes of variation of the data and the mapping ϕ represents the data generating process that yields high-dimensional data $x_i = \phi(\theta_i) \in \mathcal{M}_{\mathcal{H}}$ characterized by these modes of variation. We assume that low-dimensional representations of the observed data in the embedding space $\mathcal{Y}$, which capture as much of the metric structure of $\mathcal{M}_{\mathcal{H}}$ as possible, can be learned from the observed data. A successful embedding e then also recovers as much of the structure of the parameter space Θ as possible in the low dimensional representations $y_i = e(x_i) \in \mathcal{Y}$.

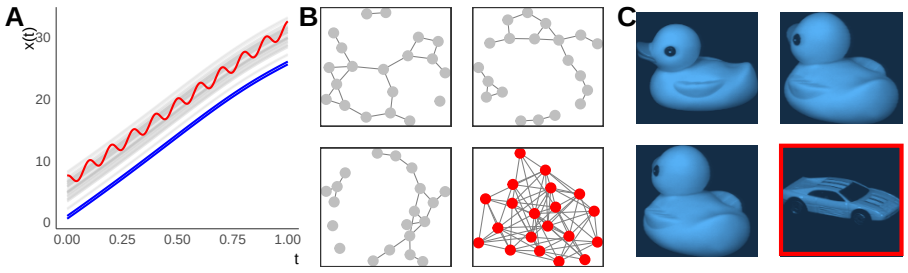

Figure 1: Example data types. A: Functional inliers (grey) with a structural outlier (red) and distributional outliers (blue). B: Graph data with a structural outlier (lower right graph). C: Image data with a structural outlier (lower right image).

In our framework, distributional outliers are defined w.r.t. minimum volume sets (Polonik, 1997) of P in this parameter space Θ:

Definition 1: *Minimum volume set*
Given a probability distribution P over (a subset of) $\mathbb{R}^d$, a minimum volume set Ω^*_α is a set that minimizes the quantile function $V(\alpha) = \inf_{C \in \mathcal{C}}\{\text{Leb}(C) : P(C) \geq \alpha\}, 0 < \alpha < 1\}$ for i.i.d. random variables in $\mathbb{R}^d$ with distribution P, $\mathcal{C}$ a class of measurable subsets in $\mathbb{R}^d$ and Lebesgue measure Leb.
So $\Omega^*_{\alpha,P}$ is the smallest region containing a probability mass of at least α. We can now define structural outliers and distributional outliers as follows:

Definition 2: *Structural and distributional outlier*
Define $\mathcal{M}_{\Theta,\phi}$ as the codomain of ϕ applied to Θ.
Define two such manifolds $\mathcal{M}_a = \mathcal{M}_{\Theta_a,\phi_a}$ and $\mathcal{M}_c = \mathcal{M}_{\Theta_c,\phi_c}$ and a data set $X \subset \mathcal{M}_a \cup \mathcal{M}_c$. W.l.o.g., let $r = \frac{|\{x_i : x_i \in \mathcal{M}_a \wedge x_i \notin \mathcal{M}_c\}|}{|\{x_i : x_i \in \mathcal{M}_c\}|} \lll 1$ be the *structural outlier ratio*, i.e. most observations are assumed to stem from $\mathcal{M}_c$. Then an observation $x_i \in X$ is

- a *structural outlier* if $x_i \in \mathcal{M}_a$ and $x_i \notin \mathcal{M}_c$ and
- a *distributional outlier* if $x_i \in \mathcal{M}_c$ and $\theta_i \notin \Omega^*_\alpha$, where Ω^*_α is defined by the density of the distribution generating Θ_a.

Figure 1 shows examples of three data types with structural outliers (in red) and some distributional outliers for the functional data example. Since distributional outliers are structurally similar to inliers, they are hard to detect visually for graph and image data, as doing so requires a lot of "normal" data to reference against and we can only display a few example observations here. As outlined, this is one reason why we again use functional data for our exposition in the following.

Summarizing the framework's crucial aspects in less technical terms, we assume that the bulk of the observations comes from a single "common" process, which generates observations in some subset $\mathcal{M}_c$, while some data might come from an "anomalous" process, which defines structurally distinct observations in a different subset $\mathcal{M}_a$. This follows standard notions in outlier detection which often assume (at least) two different data-generating processes (Dai et al., 2020; Zimek & Filzmoser, 2018). Note that this does not imply that structural outliers are in any way similar to each other: P_a could be very widely dispersed or arise from a mixture or several different distributions and/or $\mathcal{M}_a$ could consist of several unconnected components representing various

kinds of structural abnormality. The only crucial aspect is that the process from which *most* of the observations emerge yields structurally similar data. We consider settings with a structural outlier ratio $r \in [0, 0.1]$ to be suitable for outlier detection. The proportion of *distributional* outliers on $\mathcal{M}_c$, in contrast, depends only on the α-level for Ω^*_{α,P_c}. Practically speaking, neither prior knowledge about these manifolds nor specific assumptions about structural differences are necessary for our approach. The key points are that (1) structural outliers are not on the main data manifold $\mathcal{M}_c$, (2) distributional outliers are at the edges of $\mathcal{M}_c$, and (3) these properties are preserved in the embedding vectors as long as the embedding is based on an appropriate notion of distance in $\mathcal{H}$.

4 Experiments

This section lays out practical implications of the framework through experiments on several different data types, via a comprehensive qualitative and visual analysis of six examples. In addition, we provide quantitative results for six labeled data sets.

4.1 Methods

The focus of our experiments is to evaluate a general framework for outlier detection, which is motivated by geometrical considerations. With these experiments, we support the claim that the perspective induced by the framework lets us visualize, detect, and analyze outliers in a principled and canonical way. For this demonstration, we chose Multidimensional Scaling (MDS) (Cox & Cox, 2008) as our embedding method and Local Outlier Factors (LOF) (Breunig et al., 2000) as our outlier scoring method. Note that the experiments are not intended to draw conclusions about the superiority of these specific methods and other combinations of methods may be as suitable or even superior for the purpose (see for example results for Isomap in Herrmann & Scheipl (2021)).

However, more sophisticated embedding methods than MDS require tuning over multiple hyperparameters, whereas MDS has only one – the embedding dimension. Moreover, an advantage of MDS over other embedding methods is that it aims for isometric embeddings, i.e., tries to preserve all pairwise distances as closely as possible, which is crucial in particular to reflect structural outlyingness. In fact, Torgerson Multidimensional Scaling (tMDS, i.e., MDS based on L_2 distance) – that is: a simple linear embedding equivalent to standard PCA scores – seems to uncover many outlier structures sufficiently well in many data settings despite its simplicity. For similar reasons, we chose to use LOF as an outlier scoring method. This method also has a single hyperparameter, $minPts$, the number of nearest neighbors used to define a point's (local) neighborhood, which we denote as k in the following. Moreover, in contrast to many other outlier scoring methods such as one-class support vector machines (Muñoz & Moguerza, 2004) which require low-dimensional tabular data as input (i.e. which can only be applied to complex data types indirectly by using embedding vectors as feature inputs), LOF can also be applied to high-dimensional and non-tabular data directly as it only requires a distance matrix as input. Experiments on functional data have shown that LOF applied directly to a distance matrix of functional data and LOF applied to the corresponding embedding vectors yield consistent results (Herrmann & Scheipl, 2021).

Note, however, that beyond the ability to apply outlier scoring methods to low-dimensional embedding vectors of high-dimensional and/or non-tabular data, such embeddings provide additional practical value: In particular, scalar scores or ranks as provided by outlier scoring methods are not able to reflect differences between distributional and structural outliers whereas such differences become accessible and interpretable in visualizations of these embeddings.

This also points to a major caveat of the quantitative (in contrast to the qualitative) experiments, in which we use ROC-AUC to evaluate the accuracy of outlier ranks obtained with LOF with respect to the "outlier" structure defined by the different classes of labeled data. Setting one class as $\mathcal{M}_c$ and contaminating this "normal" class with observations from other classes, which are assumed to be structurally different and thus form $\mathcal{M}_a$, we obtain data sets $X \subset \mathcal{M}_c \cup \mathcal{M}_a$. Although this is a widely used approach (Campos et al., 2016; Goldstein & Uchida, 2016; Pang et al., 2018), such an evaluation only considers outliers as defined by the class labels and poor AUC values may not necessarily imply poor performance if there are observations from the "normal" data class which are (distributionally or structurally) more outlying (and thus obtain higher scores) than some of the "labeled" outliers, see also Campos et al. (2016). This is why we do not merely report potentially problematic quantitative measures (Section 4.3), and instead put more emphasis on qualitative experiments that are much closer to the way we would recommend using these methods in practical applications.

4.2 Qualitative assessment

In this section, we provide extensive qualitative analyses to demonstrate the practical relevance of the framework. First, we demonstrate that the distinction between structural and distributional outliers is preserved in embeddings using two simulated functional data sets. Secondly, using two real-world data sets – a functional and an image data set – we show that the approach can be applied flexibly to different data structures. Thirdly, we illustrate the general applicability to more general data types based on synthetic graph data and real-world curves data. In the following, all LOF results are obtained using $k = 0.75n$, where n is the number of observations.

4.2.1 Demonstrating the framework's practical implications on idealized synthetic data

Figure 2 shows two simulated functional data sets (A & B, left) and their $2\mathbf{D}$ PCA/tMDS embeddings (A & B, right). One can observe that data set A is an example with structural outliers in terms of shape and slope. This is an extended version of an example by Hernández & Muñoz (2016) and, following their notation, the two manifolds can be defined as $\mathcal{M}_a = \{x(t)|x(t) = b + 0.05t + \cos(20\pi t), b \in \mathbb{R}\} \cup \{x(t)|x(t) = (c - 0.05t) + e_t, c, e_t \in \mathbb{R}\}$ and $\mathcal{M}_c = \{x(t)|x(t) = a + 0.01t + \sin(\pi t^2), a \in \mathbb{R}\}$ with $t \in [0, 1]$ and $a \sim N(\mu = 15, \sigma = 4)$, $b \sim N(\mu = 5, \sigma = 3)$, $c \sim N(\mu = 25, \sigma = 3)$, and $e_t \sim N(\mu = 0, \sigma = 4)$. Note that the structural outliers are not all similar to each other in shape or slope, which is reflected in $\mathcal{M}_c$ being a union of two structurally different manifolds.

In contrast, data set B of various (vertically shifted) Beta-distribution densities is an example where distributional outlyingness is defined by phase – i.e. horizontal – variation and structural outlyingness by vertical shifts. The respective manifolds are defined as $\mathcal{M}_a = \{x(t)|x(t) = b+B(t, \alpha, \beta), (b, \alpha, \beta)' \in \mathbb{R}\times\mathbb{R}^+\times\mathbb{R}^+\}$ and $\mathcal{M}_c = \{x(t)|x(t) = B(t, \alpha, \beta), (\alpha, \beta)' \in \mathbb{R}^+\times\mathbb{R}^+\}$ with $t \in [0, 1]$, $\alpha, \beta \sim U[0.1, 2]$, $b \sim U[-5, 5]$ and B the density of the beta distribution. For both, we generate 100 "normal" observations from $\mathcal{M}_c$ and 10 structural outliers from $\mathcal{M}_a$, with $D = 500$ evaluation points in the first and $D = 50$ evaluation points in the latter example.

Structural outliers are clearly separated from observations on $\mathcal{M}_c$ in both cases and appear as outlying in the $2\mathbf{D}$ embeddings. Moreover, we see that distributional outliers are embedded at the periphery of $\mathcal{M}_c$. Numbers in the figures are ascending LOF score ranks of the outliers. Note that $\mathcal{M}_c \subset \mathcal{M}_a$ in data set B. Nevertheless, most structurally outlying observations from $\mathcal{M}_a$ are clearly separated in the embedding. Two structural outliers are in or very close to $\mathcal{M}_c \cup \mathcal{M}_a$ and thus appear in the main bulk of the data. The LOF scores also reflect this, as one of the distributional outliers is ranked as even more outlying.

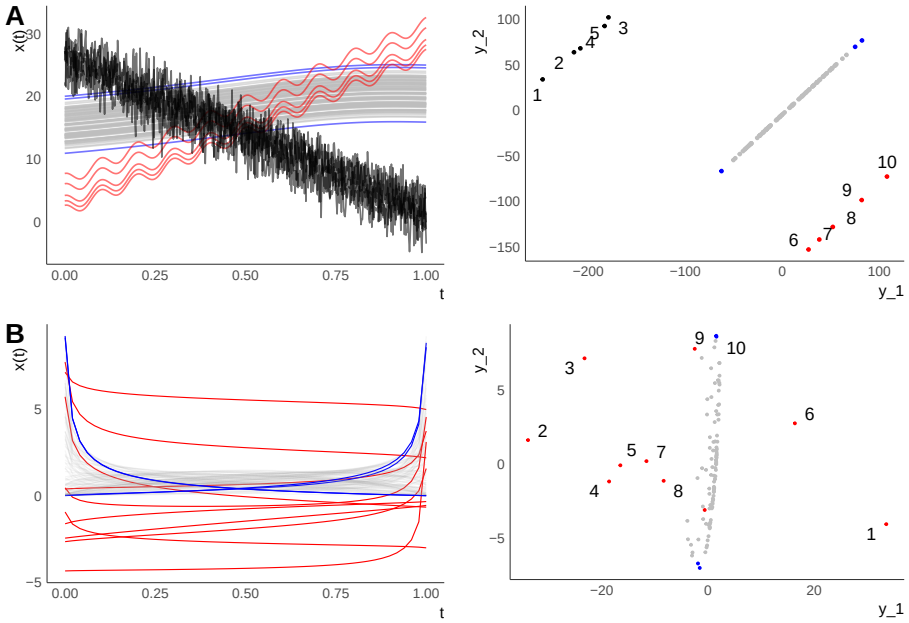

Figure 2: Simulated functional data and their $2\mathbf{D}$ embeddings. Numbered labels are ascending LOF score ranks of the outliers ($k = 0.75n$).

Summarizing, we see that in these simulated situations, practically relevant outlier sub-structure – deviations in terms of functional shape, slope, or vertical shifts – are represented accurately by low-dimensional embeddings learned from the observed high-dimensional data. In particular, structural outliers do not need to be similar to each other as Example A demonstrates. Also, note that Example B illustrates as a by-product that there can be situations where the approach yields meaningful results even though the two manifold are not completely disjoint. However, this does not necessarily hold in general. See Souvenir & Pless (2005) for an approach to disentangle intersecting manifolds. Moreover, we see that situations where distributional outliers appear "more" outlying than structural outliers are captured as well. Note that this is a crucial aspect. Although this aspect is quantified correctly by an outlier scoring method such as LOF, the two outlier types can be distinguished only if visualizations, as provided by embedding methods, are considered. Consider that evaluation of unsupervised outlier detection is often performed using a labeled data set, setting observations from one class as inliers and sampling observations from another class as outliers, and then computing binary classification performance measures such as the AUC (Campos et al., 2016; Goldstein & Uchida, 2016; Pang et al., 2018). Different class labels do not guarantee that the classes do not overlap, i.e., that the respective manifolds are disjoint in $\mathcal{H}$, nor that there are no distributional outliers appearing more outlying than structural outliers. Thus, there may be distributional outliers among the inliers which are scored as more outlying than structural outliers (see data set B) and a purely quantitative assessment is likely to mislead. Being able to create faithful visualizations of such more complex outlier structures for high-dimensional data is a crucial benefit of the proposed approach.

4.2.2 Demonstrating flexibility on real functional and image data

Of course, real-world data settings are usually more complicated than our simulated examples. First of all, real data are much more difficult to assess since the underlying manifolds are usually not directly accessible, so it is impossible to define the exact structure of the data manifolds like in the simulated examples. In addition, some data sets may not contain any clear *structural* outliers, while others may not contain any clear *distributional* outliers, or both. A crucial aspect of the approach is that, although it is based on a highly abstract conceptualization involving unobservables like the parameter space Θ and its probability measure P, it is not at all necessary to come up with any such formalization of the data generating process to put the approach into practice and obtain meaningful results, as will be demonstrated in the following.

Consider Figure 3, which shows a real functional data set of 591 ECG measurements (Dau et al., 2019; Goldberger et al., 2000) with 82 evaluation points per function, i.e. a $D = 82$ dimensional data set (A), and a sample of the COIL20 data (Nane et al., 1996) (B). It is impossible to define the exact structure of the ECG data manifold. However, the visualizations of the functions on the left-hand side suggest that there are no observations with clear structural differences in functional form: none of the curves are clearly shifted away from the bulk of the data, nor are there any curves with isolated peaks, or observations with clearly different shapes. In accordance with this observation, there is also no clearly separable structure in the embedding. However, observations that appear in low-density regions of the embedding can be regarded as distributional outliers in terms of horizontal shift, i.e., phase variation, like the three observations with the earliest minima colored in blue. This is also reflected in the scoring of the embeddings, as the observations with the lowest LOF ranks are clear distributional outliers in function space. However, the embedding provides much more complete information in this example than LOF ranks and the functional visualization alone. For example, they also pinpoint a *vertical shift* outlier in the first and last thirds of the domain (green curve, which would be hard to detect based on its functional representation alone). This apparently represents a second "dimension"

of distributional outlyingness.
The COIL20 data (Nane et al., 1996) consists of 1440 pictures (128×128, grayscale) of 20 different objects. The 72 pictures in each class depict one and the same object at different rotation angles with a picture taken at every $5°$ within $[0°, 355°]$. We use all 72 pictures of a rubber duck to represent observations from $\mathcal{M}_c$ and randomly sample 7 observations (i.e. $r \approx 0.1$) from the 72 pictures of a toy car as structural outliers from $\mathcal{M}_a$. We compute L_2 distances of the vectorized pixel intensities ($D = 128^2 = 16384$). Figure 3 B, left column, shows a sample of 6 inlier and 3 structural outlier pictures, the right column shows embeddings of all 79 images. Since the inlier data are images of a rotated object, $\mathcal{M}_c$ is the image of a one-dimensional closed and circular parameter space defining the rotation angle (c.f. Ma & Fu, 2011), i.e., other than in the ECG example substantial considerations yield at least some knowledge about the specific structure of the data manifold(s) in this case.

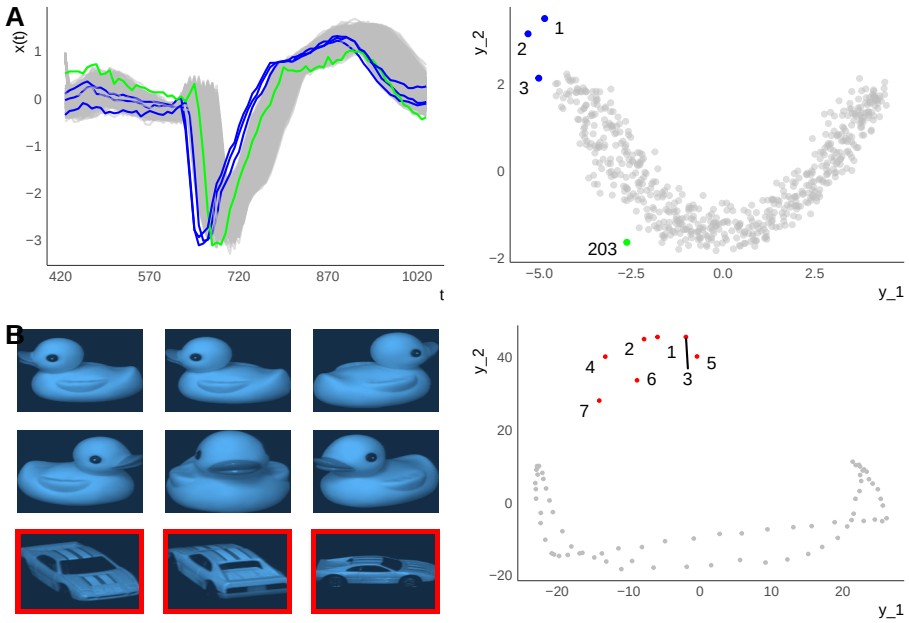

Figure 3: Real functional and image data and their $2\mathbf{D}$ tMDS embeddings. Numbered labels are ascending LOF score ranks of the outliers ($k = 0.75n$).

The $2\mathbf{D}$ embedding reflects the expected structure of our COIL20 subset very well, with clear separation of the 7 pictures of the toy car as structural outliers. In addition, the embedding of $\mathcal{M}_c$ indeed yields a closed, but not quite a circular loop, as does the embedding of the 7 rotated images from $\mathcal{M}_a$. The corresponding 3D embedding (not shown) reveals that the embeddings of the inliers lie on a rough circle folded over itself. In summary, in the ECG example there seem to be no clearly separable, structurally different outliers that could be detected with tMDS, but only distributional outliers, whereas in the COIL data there are clearly separate structural outliers, but no distributionally outlying observations. These two examples with very different intrinsic structures (single connected manifold with distributional outliers versus disconnected

manifolds without clear distributional outliers) illustrate that it is not necessary to have explicit prior knowledge about the data generating process or its outlier characteristics for the approach to work and that it is able to handle different data manifold structures flexibly and successfully.

4.2.3 Demonstrating generalizability on graph and curve data

Note that the COIL example illustrates that the framework also works in image data and that a fairly simplistic approach of computing L_2 distances between vectorized pixel intensities yields very reasonable results in this example. The framework is, however, not at all restricted to these two data types nor such a simple distance metric. Recall that the approach can be applied to any data type whatsoever as long as a suitable distance metric is available. Beyond $1\mathbf{D}$ functional and image data, the framework can also be extended to more general and complex data types, for example, graphs or $2\mathbf{D}$ curves as depicted in Figures 4. We use more specialized distance measures to show that good results can also be obtained on such data.

We simulate two structurally different classes of Erdős-Rényi graphs with 20 vertices (see Fig. 4 A). This structural difference results from different edge probabilities p_v that two given vertices of the graph are connected, setting $p_v = 0.1$ for $\mathcal{M}_c$ and $p_v = 0.4$ for $\mathcal{M}_a$. We randomly sample 100 observations from $\mathcal{M}_c$ and 10 from $\mathcal{M}_a$, i.e. $r = 0.1$, and obtain a pairwise distance matrix by computing the Frobenius distances between the graph Laplacians.

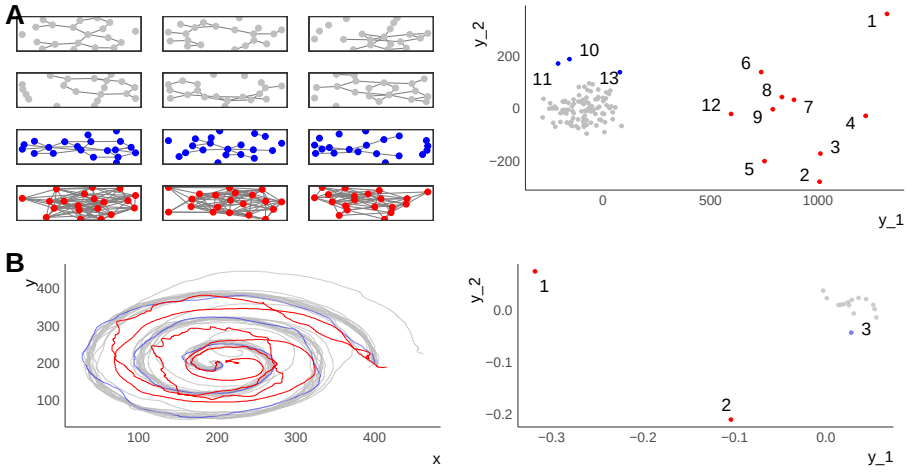

Figure 4: Curve and graph data as further examples to demonstrate the flexibility and general applicability of the approach, and their $2\mathbf{D}$ MDS embeddings based on Frobenius (graphs) and Elastic shape distances (curves). Numbered labels are ascending LOF score ranks of the outliers ($k = 0.75n$).

The curves data (Fig. 4 B) consists of spiral curve drawings from an Archimedes spiral-drawing test that is used to diagnose patients with Parkinson's disease (Alty et al., 2017; Steyer et al., 2021). Taking data from the dynamic version of the test (Isenkul et al., 2014), we use 15 curves drawn by healthy controls not suffering from Parkinson's disease and two curves drawn by Parkinson patients to represent potential structural outliers, where each curve is evaluated on 200 points. Previous investigations have shown that an elastic shape distance is better suited than L_2 distances to discriminate between the two groups (Steyer et al., 2021).

So, in contrast to the previous examples, we use more specialized distance measures to capture the relevant structures in these settings. This illustrates that the approach is not only flexible with respect to the actual structure present in a given data set as demonstrated in the previous section but that it is also very generally applicable to a variety of data types. The approach can be used for any kind of data simply by defining an appropriate (data-specific) distance measure. In both, the embeddings of the graphs, as well as the embeddings of the curves, structurally different observations (in red) are clearly separated from the observations on $\mathcal{M}_c$. This is also reflected by their LOF scores. Moreover, in both settings, there are observations from $\mathcal{M}_c$ (in blue) which appear in peripheral, sparser regions of the "normal" data and thus can be considered distributional outliers. Note that it is not always immediately obvious on the level of the original data why observations appear distributionally outlying. For example, in the graph data, note that other than in previous examples (e.g. Fig. 2 A) comparing them to a few inliers does not reveal a striking difference at first (in contrast to the structural outliers!): Figure 4 A, left column, shows six inlier graphs in the 1st and 2nd row, the three distributional outlier graphs in the 3rd row, and three structural outlier graphs in the 4th row.
Nevertheless, the embedding vectors and their LOF ranks indicate that the distributionally outlying observations have obtained some specific characteristics setting them apart from most inlying observations. For example, further analysis reveals that the graph with LOF rank 11 contains the node with maximum connectedness of all nodes in all inlier graphs. Its degree is 8 (i.e., it is directly connected to 8 other nodes), while the average of the maximum degree in the graphs on $\mathcal{M}_c$ is just 4.39. In contrast, the graph with LOF rank 13 contains 8 isolated nodes of degree 0, while the average number of nodes with degree 0 is only 2.47 on $\mathcal{M}_c$. The respective values of the graph with LOF rank 10 are above the upper quartile for both of these metrics, with 4 unconnected nodes and a maximally connected node with degree 6.

4.3 Quantitative assessment

In order to provide less subjective experimental results, we assess the approach quantitatively, using labeled data with at least two classes. For each data set, we consider four outlier ratios $r \in \{0.01, 0.025, 0.5, 0.1\}$. Setting one class as $\mathcal{M}_c$, with $n_{in} = |\mathcal{M}_c|$, and contaminating this "normal" class with $n_{out} = r \cdot n_{in}$ "structural" outliers from other classes, which form $\mathcal{M}_a$, we obtain data sets $X \subset \mathcal{M}_c \cup \mathcal{M}_a$ with $n = n_{in} + n_{out}$. For each setting, we repeat the contamination process 50 times, sampling outliers at random from $\mathcal{M}_a$. Based on outlier ranks computed with LOF, we use ROC-AUC as a performance measure and report the mean AUCs over the 50 replications for each combination of settings. Note that we only use the labels of the "structural" outliers for computing this performance measure, not for the unsupervised learning of the embeddings themselves. For all data sets considered in this section, plots of typical embeddings for $r = 0.05$ can be found in Figures 5 and 6 in appendix A. We consider three additional functional data sets for this experiment: *dodgers* (Dau et al., 2019), a set of times series of daily traffic close to Dodgers Stadium, with days on weekends forming $\mathcal{M}_a$ and weekdays forming $\mathcal{M}_c$; *phoneme* (Febrero-Bande & Oviedo de la Fuente, 2012), discretized log-periodograms of five different phonemes, with phoneme "dcl" forming $\mathcal{M}_a$ and phonemes "sh", "iy", "aa", and "ao" forming $\mathcal{M}_c$; *starlight* (Dau et al., 2019; Rebbapragada et al., 2009), phase-aligned light curves of Eclipsing Binary, Cepheid, and RR Lyrae stars, the first forming $\mathcal{M}_a$ and the latter two forming $\mathcal{M}_c$. All results are based on simple, linear tMDS/PCA embeddings with the LOF algorithm applied to the resulting $2\mathbf{D}$ embedding vectors.
In addition, we consider three tabular data sets, two real and one simulated. This includes the well-known Iris data (Anderson, 1935; Fisher, 1936) where class *Setosa* forms $\mathcal{M}_c$ and the other two classes $\mathcal{M}_a$. Moreover, we use the Wiscon Breast Cancer (wbc) data (Street et al., 1993)

Table 1: Mean ROC-AUC values over 50 replications based on the ranks as assigned by LOF. Each data set consists of n observations, n_{in} from $\mathcal{M}_c$ and $n_{out} = n_{in} \cdot r$ from $\mathcal{M}_a$. $\mathcal{M}_a$ and $\mathcal{M}_c$ are defined by classes of the original labeled data sets. D is the dimensionality of a data set (i.e, evaluations per function for functional data) and k the number of nearest neighbors used in the LOF algorithm. A: Functional data. B: Tabular data.

A		dodgers $n_{in} = 97,\ D = 289$				phoneme $n_{in} = 400,\ D = 150$				starlight $n_{in} = 6656,\ D = 1025$			
	k	$0.01n$	$0.1n$	$0.75n$	$0.9n$	$0.01n$	$0.1n$	$0.75n$	$0.9n$	$0.01n$	$0.1n$	$0.75n$	$0.9n$
$r: 1.0\%$		0.78	0.98	0.96	0.96	0.78	1.00	0.99	0.99	0.96	1.00	0.69	0.78
$r: 2.5\%$		0.62	0.97	0.96	0.96	0.54	1.00	0.99	0.99	0.55	1.00	0.88	0.88
$r: 5.0\%$		0.59	0.97	0.96	0.96	0.56	0.99	0.99	0.99	0.53	1.00	0.92	0.92
$r: 10\%$		0.54	0.84	0.97	0.96	0.57	0.75	0.99	0.99	0.56	0.98	0.95	0.87
B		iris $n_{in} = 50,\ D = 4$				wisconsin breast cancer $n_{in} = 357,\ D = 30$				simulated data $n_{in} = 750,\ D = 1000$			
	k	$0.01n$	$0.1n$	$0.75n$	$0.9n$	$0.01n$	$0.1n$	$0.75n$	$0.9n$	$0.01n$	$0.1n$	$0.75n$	$0.9n$
$r: 1.0\%$		1.00	1.00	1.00	1.00	0.76	0.96	0.94	0.89	0.66	1.00	1.00	1.00
$r: 2.5\%$		0.71	1.00	1.00	1.00	0.64	0.97	0.95	0.92	0.56	1.00	1.00	1.00
$r: 5.0\%$		0.52	1.00	1.00	1.00	0.60	0.97	0.94	0.91	0.58	1.00	1.00	1.00
$r: 10\%$		0.61	0.69	1.00	1.00	0.58	0.96	0.94	0.92	0.56	1.00	1.00	1.00

as provided by the UCI Machine Learning repository (Dua & Graff, 2017). This tabular data set comprises 30 features containing information about the cell nuclei of breast tissue and has been used by Goldstein & Uchida (2016) for outlier detection before. Following their approach, the healthy patients form $\mathcal{M}_c$ and patients with malignant status form $\mathcal{M}_a$. Yet, other than Goldstein & Uchida (2016), we do not fix outliers to the first 10 observations from the latter class but – as outlined – repeatedly sample outliers at random from $\mathcal{M}_a$. Finally, we include a simple simulated example where $\mathcal{M}_c = \{x : x \sim \mathcal{N}_{1000}(\mathbf{0}, \Sigma)\}$ and $\mathcal{M}_a = \{x : x \sim \mathcal{N}_{1000}(\mathbf{1}, \Sigma)\}$, $\Sigma = \text{diag}(\mathbf{1})$. That is, 1000-dimensional data with observations sampled from two multivariate normal distributions where the class difference stems from the difference in the mean vectors, $\mathbf{0}$ for $\mathcal{M}_c$ and $\mathbf{1}$ for $\mathcal{M}_a$.

The results depicted in Table 1 show that outlier detection does not need to be specifically challenging in nominally high-dimensional data. In each of the data sets, which have very different numbers of observations and numbers of dimensions, high ROC-AUC ≥ 0.95 can be achieved for all considered outlier ratios r. This indicates that most of the observations from $\mathcal{M}_a$ indeed appear to be outlying in the embedding space and thus obtain high LOF scores. Furthermore, as in the qualitative analysis, a global setting of $k = 0.75n$ seems to be a reasonable default for the LOF algorithm. Only for $r = 0.01, 0.025$ in the starlight data, we see a large improvement (AUC $= 1.00$) with $k = 0.1n$. For small $r < 0.1$, in all other settings the achieved ROC-AUC is very robust against changes in this tuning parameter.

5 Discussion

5.1 Summary

We propose a geometrically motivated framework for outlier detection, which exploits the metric structure of a (possibly high-dimensional) data set and provides a mathematically precise distinction between *distributional* outliers and *structural* outliers. Experiments show that the outlier structure of high-dimensional and non-tabular data can be detected, visualized, and quantified using established manifold learning methods and standard outlier scoring. The decisive advantage of our framework from a theoretical perspective is that the resulting embeddings make

subtle but important properties of outlier structure explicit and – even more importantly – that these properties are made accessible based on visualizations of the embeddings. From a more practical perspective, our proposal requires no prior knowledge nor any specific assumptions about the actual data structure in order to work, an important aspect since data generating processes are usually inaccessible. This is highly relevant in practice, in particular since a well-established, computationally cheap combination of widely used and fairly simple methods like (t)MDS and LOF proved to be a strong baseline that yields fairly reliable results without the need for tuning hyperparameters. In addition, the proposed framework has several more general conceptual implications for outlier detection which will be summarized in the following.

5.2 Implications

Outlier taxonomy We propose a clear taxonomy to distinguish between frequently interchangeably used terms *anomalies* and *outliers* in a canonical way: we regard *anomalies* as observations from a different data generating process than the majority of the data (i.e. as observations that are on $\mathcal{M}_a$ but not on $\mathcal{M}_c$), which can be more precisely identified as *structural* outliers. Recall that Zimek and Filzmoser (2018, p. 10) refer to such observations as "real" outliers that need to be distinguished from "observations which are in the extremes of the model distribution". On the other hand, regarding *outliers* as observations from low-density regions of the underlying "normal" data manifold $\mathcal{M}_c$, they can be more precisely identified as *distributional* outliers. Based on our reading of the literature, this distinction is usually not made explicit. Since there is rarely a practical reason to assume that a given data set contains only *distributional* or only *structural* outliers, some of the confusion surrounding the topic (Goldstein & Uchida, 2016; Unwin, 2019; Zimek & Filzmoser, 2018) might be because such conceptual differences have not been made sufficiently clear. As outlined, the concept of structural difference is very general. For example, structural differences in functional data may appear as shape anomalies in data mainly characterized by vertical shift variation (see Fig. 1 A) or as vertical shift anomalies in data dominated by shape variation, as phase anomalies in data with magnitude variation or magnitude anomalies in data with phase variation, etc.

In real unlabeled data, there may not always be a clear distinction between somewhat structurally anomalous observations with "off-manifold" embeddings and merely distributionally outlying observations with embeddings on the periphery of the data manifold, as in the ECG data in Figure 3 A. Nevertheless, the theoretical distinction between these two kinds of outliers adds conceptual clarity even if the practical application of the categories may not be straightforward.

Curse of dimensionality As outlined in section 2.1, outlier detection is often reported to suffer from the curse of dimensionality. For example, Goldstein & Uchida (2016) show that most outlier detection methods under consideration break down or perform poorly in a data set with 400 dimensions and conclude that unsupervised outlier detection is not possible in such high dimensions. Some [Aggarwal (2017); e.g.] attribute this to the fundamental problem that distance functions can lose their discriminating power in high dimensions (Beyer et al., 1999), which is linked to the concentration of measure effect (Pestov, 2000). However, this effect occurs only under fairly specific conditions (Zimek et al., 2012), which means that outlier detection does not have to be affected by the curse of dimensionality: In addition to the effects of dependency structures and signal-to-noise ratios (Zimek et al., 2012), the necessary conditions for concentration of measure are not fulfilled if the intrinsic dimensionality of the data is smaller than the actually observed dimensionality, or if the data is distributed in clusters that are relatively well separable (Beyer et al., 1999). Exactly these two characteristics are reflected in our framework in the form of (1) the manifold assumption, which implies low-ish intrinsic dimensionality, and (2) the assumption that structural outliers come from different manifolds than the rest of the data, i.e., from different

"clusters" in $\mathcal{H}$. This has two important consequences: First of all, the geometric perspective our framework is based on makes these important aspects for outlier detection in high-dimensional data explicit, while a purely probabilistic perspective obscures them. Secondly, it mitigates many of the problems associated with high-dimensional outlier detection: any outlier detection method that performs well in low dimensions becomes – in principle – applicable in nominally high-dimensional and/or complex non-tabular data when applied to suitable low-dimensional embedding coordinates. In addition, our results show that outlier sub-structure, specifically the differences between distributional and structural outliers, can be detected and visualized with manifold methods. This opens new possibilities for descriptive and exploratory analyses:

Visualizability of outlier characteristics If the embeddings provided by manifold methods are restricted to two or three dimensions, they also provide easily accessible visualizations of the data. In fact, manifold learning is often used in applications specifically to find two- or three-dimensional visualizations reflecting the essential intrinsic structure of the high-dimensional data as faithfully as possible. Consequently, structural and distributional outliers, which are rather glaring data characteristics if the manifolds are well separable, can often be separated clearly even in two- or three-dimensional representations as long as the embedding is (approximately) isometric with respect to a suitable dissimilarity measure. This is specifically important for complex non-tabular or high-dimensional data types such as images or graphs, where at most a few observations can be visualized and perceived simultaneously. In the same vein, substructures and notions of data depth are reflected in the embeddings, making the approach also useful as an exploration tool for settings with unclear structure.

Generalizability Since the central building block of the proposed framework is to capture the metric structure of data sets using distance measures, the framework is very general and applicable to any data type for which distance metrics are available. In Section 4.2, we illustrated this generalizability using high-dimensional as well as non-tabular data; in particular, we applied it to functional, curve, graph, and image data. This also makes the framework very flexible as one can make use of non-standard and customized dissimilarity measures to emphasize the relevant structural differences in specific situations based on domain knowledge: Representing image data as vectors of pixel intensities, we computed distances between those vectors, for example. Dissimilarities between different graphs were captured, for example, by constructing their graph Laplacians and computing Frobenius distances between them, and we used a specific elastic depth distance for the spiral curve data as suggested by earlier results in Steyer et al. (2021).

5.3 Limitations and outlook

If in an exploratory setting, observations appear clearly separated in the (first few) embedding dimensions, we can be sure they are structural outliers. Note that if D-dimensional data actually live in a d'-dimensional subspace, constructing a d'-dimensional embedding with MDS based on L_2 distances will lead to an embedding with a distance matrix exactly matching the distance matrix in the D-dimensional space, i.e. MDS is isometric by design. If other than L_2 distances are used this still holds approximately (Young & Householder, 1938; see also Cox & Cox, 2008; Torgerson, 1952). Note that this is an important difference from many other dimension reduction methods. For example, UMAP is based on a local connectivity constraint (McInnes et al., 2020) which ensures that each point is at least connected to its nearest neighbor and which runs counter to a reliable embedding of structural outliers. In addition, more sophisticated methods require parameter tuning for any given setting, which is inherently difficult for unsupervised tasks, and it is not always clear how to tune other embedding methods so that they yield (approximately) isometric embeddings.

Clear structural outliers are the source of large variation in data sets with low intrinsic dimension-

ality. Since MDS embedding dimensions are sorted according to the decreasing variation, they will be reflected in the first few embedding dimensions. It may be that some of the distributional outliers are masked due to projection if the embedding dimension d is smaller than d' but following Zimek & Filzmoser (2018) we consider faithfully reflecting structural outliers more important. However, inliers, i.e. observations on $\mathcal{M}_c$, may show large "within class" variation and/or may be spread over several disconnected clusters in some situations. For example, object images on $\mathcal{M}_c$, which are structurally similar in terms of the depicted objects' shape, may vary in rotation, scale, or location, and may have different colors or textures. In functional data, observations on $\mathcal{M}_c$ may show phase and amplitude variation and form clusters due to different shapes. In such settings, $\mathcal{M}_c$ can yield complex substructure and highly dispersed observations and it may be hard to distinguish whether separable structures observed in embeddings are due to groups of homogeneous structural outliers or due to multimodality in $\mathcal{M}_c$ in which some modes are sparsely sampled. Moreover, in such cases, the dispersion of $\mathcal{M}_c$ accounts for large parts of the data's variability, and two- or three-dimensional MDS embeddings may not be sufficient to also faithfully represent structural outliers, since MDS embedding vectors are sorted decreasingly by explained "variance". However, this does not mean that structural outliers are not necessarily separable. Instead, they appear as outliers in higher embedding dimensions, requiring higher order embeddings to reflect the outlier structure. That means, if in an exploratory setting, there are no clearly separated observations in the (first few) embedding dimensions, there are either no clear structural outliers or they appear in later embedding dimensions if there are sources in M_c that induce more variation than the structural outlier. For example, objects in images may be structurally different in texture but not in color, orientation, and scale. In such a case – all observations differ in color, orientation, and scale but only some observations in texture –, these other aspects can induce large variation within observations on M_c, and the structural difference in texture is loaded on latter embedding dimensions. In such a situation, one can use scatterplot matrices and *Scagnostics* (scatterplot diagnostics, Wilkinson et al., 2005) for visual inspection. In addition, one can check out the kurtosis of the LOF scores in different embedding dimensions or *high contrast subspaces for density-based outlier ranking* (HiCS, Keller et al., 2012), to find pairs of dimensions that are "interesting" in terms of structural outliers. Moreover, techniques from multi-view learning such as "distance-learning from multiple views" may likely yield better results, because different structures (e.g. structure induced by color vs structure induced by texture) should be "treated separately as they are *semantically* different" (Zimek & Vreeken, 2015, p. 128). Note, however, that suitable inductive biases can also be brought to bear in our framework fairly easily. If substantial considerations suggest that specific structural aspects are important, specifying dissimilarity metrics focused on these aspects allows to emphasize the relevant differences. For example, if isolated outliers in functional data (i.e. functions which yield outlying behavior only over small parts of the domain such as isolated peaks) are of most interest, higher order L_p metrics such as L_{10} will be much more sensitive to such structural differences than general L_2 distances. If phase variation should be ignored, the unnormalized L_1-Wasserstein or the Dynamic Time Warping (DTW) distance can be used. Such problem-specific distance measures can reduce the number of MDS embedding dimensions necessary for faithful embeddings of structural outliers (Herrmann & Scheipl, 2021). In future work, we will investigate these aspects and possible extensions w.r.t. to multi-view learning approaches. Moreover, we will elaborate more on the specifics of other data types, in particular, image data.

6 Conclusion

In conclusion, our illustration suggests that the proposed geometric conceptualization, which distinguishes *distributional* and *structural* outliers on a general level, provides a more precise terminology and shows that outlier detection in high-dimensional and complex non-tabular data does need to be specifically challenging per se. Convincing results could be achieved in a wide range of settings and data types by a combination of the simple methods MDS for dimension reduction and visualization and LOF for outlier scoring. We hope that the proposed framework contributes to a better understanding of unsupervised outlier detection and provides some guidance to practitioners as well as methodological researchers in this regard.

Funding

This work has been funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibility for its content.

Acknowledgment

The authors thank Almond Stöcker for his helpful advice regarding the spiral curve data.

Conflict of interest

The authors have declared no conflicts of interest for this article.

Data availability statement

The data and code to reproduce the findings of this study are openly available on GitHub at: https://github.com/HerrMo/geo-outlier-framework

References

Aggarwal, C. C. (2017). *Outlier analysis* (2nd ed.). Springer. https://doi.org/10.1007/978-3-319-47578-3

Aggarwal, C. C., & Yu, P. S. (2001). Outlier detection for high dimensional data. *SIGMOD Rec.*, *30*(2). https://doi.org/10.1145/376284.375668

Ali, M., Jones, M. W., Xie, X., & Williams, M. (2019). TimeCluster: Dimension reduction applied to temporal data for visual analytics. *The Visual Computer*, *35*(6), 1013–1026. https://doi.org/10.1007/s00371-019-01673-y

Alty, J., Cosgrove, J., Thorpe, D., & Kempster, P. (2017). How to use pen and paper tasks to aid tremor diagnosis in the clinic. *Practical Neurology*, *17*(6), 456–463. https://doi.org/10.1136/practneurol-2017-001719

Anderson, E. (1935). The irises of the gaspé peninsula. *Bull Am Iris Soc*, *59*, 2–5.

Azcorra, A., Chiroque, L. F., Cuevas, R., Anta, A. F., Laniado, H., Lillo, R. E., Romo, J., & Sguera, C. (2018). Unsupervised scalable statistical method for identifying influential users in online social networks. *Scientific Reports*, *8*(1), 1–7. https://doi.org/10.1038/s41598-018-24874-2

Beckman, R. J., & Cook, R. D. (1983). Outlier ... ... s. *Technometrics*, *25*(2), 119–149. https://doi.org/10.1080/00401706.1983.10487840

Belkin, M., & Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, *15*(6), 1373–1396. https://doi.org/10.1162/089976603321780317

Beyer, K., Goldstein, J., Ramakrishnan, R., & Shaft, U. (1999). When is "nearest neighbor" meaningful? In C. Beeri & P. Buneman (Eds.), *Database theory — ICDT'99* (pp. 217–235). Springer Berlin Heidelberg. https://doi.org/10.1007/3-540-49257-7_15

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: Identifying density-based local outliers. *SIGMOD Rec.*, *29*(2), 93–104. https://doi.org/10.1145/335191.335388

Campos, G. O., Zimek, A., Sander, J., Campello, R. J., Micenková, B., Schubert, E., Assent, I., & Houle, M. E. (2016). On the evaluation of unsupervised outlier detection: Measures, datasets, and an empirical study. *Data Mining and Knowledge Discovery*, *30*(4), 891–927. https://doi.org/10.1007/s10618-015-0444-8

Clémençon, S., & Jakubowicz, J. (2013). Scoring anomalies: A M-estimation formulation. In C. M. Carvalho & P. Ravikumar (Eds.), *Proceedings of the sixteenth international conference on artificial intelligence and statistics* (Vol. 31, pp. 659–667). PMLR. https://proceedings.mlr.press/v31/clemencon13a.html

Coifman, R. R., & Lafon, S. (2006). Diffusion maps. *Applied and Computational Harmonic Analysis*, *21*(1), 5–30. https://doi.org/10.1016/j.acha.2006.04.006

Cox, M. A. A., & Cox, T. F. (2008). Multidimensional Scaling. In C. Chen, W. Härdle, & A. Unwin (Eds.), *Handbook of Data Visualization* (pp. 315–347). Springer. https://doi.org/10.1007/978-3-540-33037-0_14

Dai, W., Mrkvička, T., Sun, Y., & Genton, M. G. (2020). Functional outlier detection and taxonomy by sequential transformations. *Computational Statistics & Data Analysis*, *149*, 106960. https://doi.org/10.1016/j.csda.2020.106960

Dau, H. A., Bagnall, A., Kamgar, K., Yeh, C.-C. M., Zhu, Y., Gharghabi, S., Ratanamahatana, C. A., & Keogh, E. (2019). The UCR time series archive. *IEEE/CAA Journal of Automatica Sinica*, *6*(6), 1293–1305. https://doi.org/10.1109/JAS.2019.1911747

Dua, D., & Graff, C. (2017). *UCI machine learning repository*. University of California, Irvine, School of Information; Computer Sciences. http://archive.ics.uci.edu/ml

Febrero-Bande, M., & Oviedo de la Fuente, M. (2012). Statistical computing in functional data analysis: The R package fda.usc. *Journal of Statistical Software*, *51*(4), 1–28. https://doi.org/10.18637/jss.v051.i04

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Ann Eugen*, *7*(2), 179–188.

Fritsch, V., Varoquaux, G., Thyreau, B., Poline, J.-B., & Thirion, B. (2012). Detecting outliers in high-dimensional neuroimaging datasets with robust covariance estimators. *Medical Image Analysis*, *16*(7), 1359–1370. https://doi.org/10.1016/j.media.2012.05.002

Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, *101*(23), e215–e220. https://doi.org/10.1161/01.CIR.101.23.e215

Goldstein, M., & Uchida, S. (2016). A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. *PloS One*, *11*(4), e0152173. https://doi.org/10.1371/journal.pone.0152173

Guan, S., & Loew, M. (2021). A Novel Intrinsic Measure of Data Separability. *arXiv:2109.05180 [Cs, Math, Stat]*. http://arxiv.org/abs/2109.05180

Hernández, N., & Muñoz, A. (2016). Kernel Depth Measures for Functional Data with Application to Outlier Detection. In A. E. P. Villa, P. Masulli, & A. J. Pons Rivero (Eds.), *Artificial*

neural networks and machine learning – ICANN 2016 (pp. 235–242). Springer, Cham. https://doi.org/10.1007/978-3-319-44781-0_28

Herrmann, M., & Scheipl, F. (2021). A geometric perspective on functional outlier detection. *Stats*, *4*(4), 971–1011. https://doi.org/10.3390/stats4040057

Isenkul, M., Sakar, B., Kursun, O., et al. (2014). Improved spiral test using digitized graphics tablet for monitoring parkinson's disease. *The 2nd International Conference on e-Health and Telemedicine (ICEHTM-2014)*, *5*, 171–175.

Kamalov, F., & Leung, H. H. (2020). Outlier detection in high dimensional data. *Journal of Information & Knowledge Management*, *19*(01), 2040013. https://doi.org/10.1142/S0219649220400134

Kandanaarachchi, S., & Hyndman, R. J. (2020). Dimension reduction for outlier detection using DOBIN. *Journal of Computational and Graphical Statistics*, 1–16. https://doi.org/10.1080/10618600.2020.1807353

Keller, F., Muller, E., & Bohm, K. (2012). HiCS: High Contrast Subspaces for Density-Based Outlier Ranking. *2012 IEEE 28th International Conference on Data Engineering*, 1037–1048. https://doi.org/10.1109/ICDE.2012.88

Lee, J. A., & Verleysen, M. (2007). *Nonlinear Dimensionality Reduction* (1st ed.). Springer Science & Business Media. https://doi.org/10.1007/978-0-387-39351-3

Loperfido, N. (2020). Kurtosis-based projection pursuit for outlier detection in financial time series. *The European Journal of Finance*, *26*(2-3), 142–164. https://doi.org/10.1080/1351847X.2019.1647864

Ma, Y., & Fu, Y. (Eds.). (2011). *Manifold learning theory and applications* (1st ed.). CRC press. https://doi.org/doi.org/10.1201/b11431

Maaten, L. van der, & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(86), 2579–2605. http://jmlr.org/papers/v9/vandermaaten08a.html

Marques, H. O., Campello, R. J., Sander, J., & Zimek, A. (2020). Internal evaluation of unsupervised outlier detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *14*(4), 1–42. https://doi.org/10.1145/3394053

McInnes, L., Healy, J., & Melville, J. (2020). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv:1802.03426 [Cs, Stat]*. http://arxiv.org/abs/1802.03426

McInnes, L., Healy, J., & Melville, J. (2018). *UMAP: Uniform manifold approximation and projection for dimension reduction*. arXiv. https://doi.org/10.48550/ARXIV.1802.03426

Mordohai, P., & Medioni, G. (2010). Dimensionality Estimation, Manifold Learning and Function Approximation using Tensor Voting. *Journal of Machine Learning Research*, *11*(12), 411–450. http://jmlr.org/papers/v11/mordohai10a.html

Muñoz, A., & Moguerza, J. M. (2004). One-class support vector machines and density estimation: The precise relation. In A. Sanfeliu, J. F. Martínez Trinidad, & J. A. Carrasco Ochoa (Eds.), *Progress in pattern recognition, image analysis and applications* (pp. 216–223). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-30463-0_27

Nane, S., Nayar, S., & Murase, H. (1996). Columbia object image library: COIL-20. *Dept. Comp. Sci., Columbia University, New York, Tech. Rep*.

Navarro-Esteban, P., & Cuesta-Albertos, J. A. (2021). High-dimensional outlier detection using random projections. *TEST*, *30*(4), 908–934. https://doi.org/10.1007/s11749-020-00750-y

Niyogi, P., Smale, S., & Weinberger, S. (2011). A Topological View of Unsupervised Learning from Noisy Data. *SIAM J. Comput.*, *40*, 646–663. https://doi.org/10.1137/090762932

Pang, G., Cao, L., Chen, L., & Liu, H. (2018). Learning representations of ultrahigh-dimensional data for random distance-based outlier detection. *Proceedings of the 24th ACM SIGKDD*

International Conference on Knowledge Discovery & Data Mining, 2041–2050. https://doi.org/10.1145/3219819.3220042

Pestov, V. (2000). On the geometry of similarity search: Dimensionality curse and concentration of measure. *Information Processing Letters*, *73*(1-2), 47–51. https://doi.org/10.1016/S0020-0190(99)00156-8

Polonik, W. (1997). Minimum volume sets and generalized quantile processes. *Stochastic Processes and Their Applications*, *69*(1), 1–24. https://doi.org/10.1016/S0304-4149(97)00028-8

Ramsay, J. O., & Silverman, B. W. (2005). *Functional data analysis* (2nd ed). Springer. https://doi.org/10.1007/b98888

Rebbapragada, U., Protopapas, P., Brodley, C. E., & Alcock, C. (2009). Finding anomalous periodic time series. *Machine Learning*, *74*(3), 281–313. https://doi.org/10.1007/s10994-008-5093-3

Ren, H., Chen, N., & Zou, C. (2017). Projection-based outlier detection in functional data. *Biometrika*, *104*(2), 411–423. https://doi.org/10.1093/biomet/asx012

Ro, K., Zou, C., Wang, Z., & Yin, G. (2015). Outlier detection for high-dimensional data. *Biometrika*, *102*(3), 589–599. https://doi.org/10.1093/biomet/asv021

Rousseeuw, P. J., & Leroy, A. M. (2005). *Robust regression and outlier detection*. John Wiley & Sons. https://doi.org/10.1002/0471725382

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*(5500), 2323–2326. https://doi.org/10.1126/science.290.5500.2323

Scott, C. D., & Nowak, R. D. (2006). Learning minimum volume sets. *The Journal of Machine Learning Research*, *7*(24), 665–704. http://jmlr.org/papers/v7/scott06a.html

Souvenir, R., & Pless, R. (2005). Manifold clustering. *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, 648–653 Vol. 1. https://doi.org/10.1109/ICCV.2005.149

Steyer, L., Stöcker, A., & Greven, S. (2021). *Elastic analysis of irregularly or sparsely sampled curves*. arXiv. https://doi.org/10.48550/ARXIV.2104.11039

Street, W. N., Wolberg, W. H., & Mangasarian, O. L. (1993). Nuclear feature extraction for breast tumor diagnosis. *Biomedical Image Processing and Biomedical Visualization*, *1905*, 861–870. https://doi.org/10.1117/12.148698

Tenenbaum, J. B., Silva, V. de, & Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, *290*(5500), 2319–2323. https://doi.org/10.1126/science.290.5500.2319

Thudumu, S., Branch, P., Jin, J., & Singh, J. J. (2020). A comprehensive survey of anomaly detection techniques for high dimensional big data. *Journal of Big Data*, *7*(1), 1–30. https://doi.org/10.1186/s40537-020-00320-x

Toivola, J., Prada, M. A., & Hollmén, J. (2010). Novelty detection in projected spaces for structural health monitoring. In P. R. Cohen, N. M. Adams, & M. R. Berthold (Eds.), *Advances in intelligent data analysis IX* (pp. 208–219). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-13062-5_20

Torgerson, W. S. (1952). Multidimensional scaling: I. Theory and method. *Psychometrika*, *17*(4), 401–419. https://doi.org/10.1007/BF02288916

Unwin, A. (2019). Multivariate outliers and the O3 plot. *Journal of Computational and Graphical Statistics*, *28*(3), 635–643. https://doi.org/10.1080/10618600.2019.1575226

Wilkinson, L., Anand, A., & Grossman, R. (2005). Graph-theoretic scagnostics. *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, 157–164. https://doi.org/10.1109/INFVIS.2005.1532142

Xie, W., Kurtek, S., Bharath, K., & Sun, Y. (2017). A Geometric Approach to Visualization of Variability in Functional data. *Journal of the American Statistical Association*, *112*(519), 979–993. https://doi.org/10.1080/01621459.2016.1256813

Xu, X., Liu, H., Li, L., & Yao, M. (2018). A comparison of outlier detection techniques for high-dimensional data. *International Journal of Computational Intelligence Systems*, *11*(1), 652–662. https://doi.org/10.2991/ijcis.11.1.50

Young, G., & Householder, A. S. (1938). Discussion of a set of points in terms of their mutual distances. *Psychometrika*, *3*(1), 19–22. https://doi.org/10.1007/BF02287916

Zhang, J., & Zulkernine, M. (2006). Anomaly based network intrusion detection with unsupervised outlier detection. *2006 IEEE International Conference on Communications*, *5*, 2388–2393. https://doi.org/10.1109/ICC.2006.255127

Zimek, A., & Filzmoser, P. (2018). There and back again: Outlier detection between statistical reasoning and data mining algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(6), e1280. https://doi.org/10.1002/widm.1280

Zimek, A., Schubert, E., & Kriegel, H.-P. (2012). A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *5*(5), 363–387. https://doi.org/10.1002/sam.11161

Zimek, A., & Vreeken, J. (2015). The blind men and the elephant: On meeting the problem of multiple truths in data from clustering and pattern mining perspectives. *Machine Learning*, *98*(1), 121–155. https://doi.org/10.1007/s10994-013-5334-y

A Example visualizations of the data used in the quantitative experiments

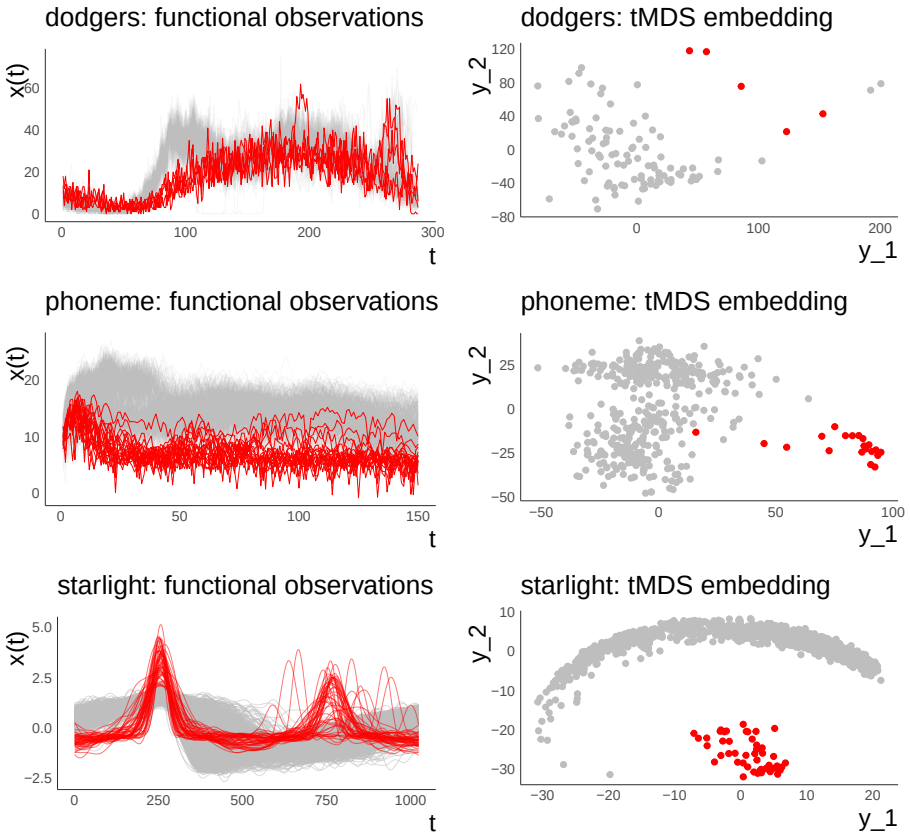

Figure 5: Plots to Table 1 A: Functional data and tMDS embeddings. Inlier class in grey, outlier class in red. $r = 0.05$

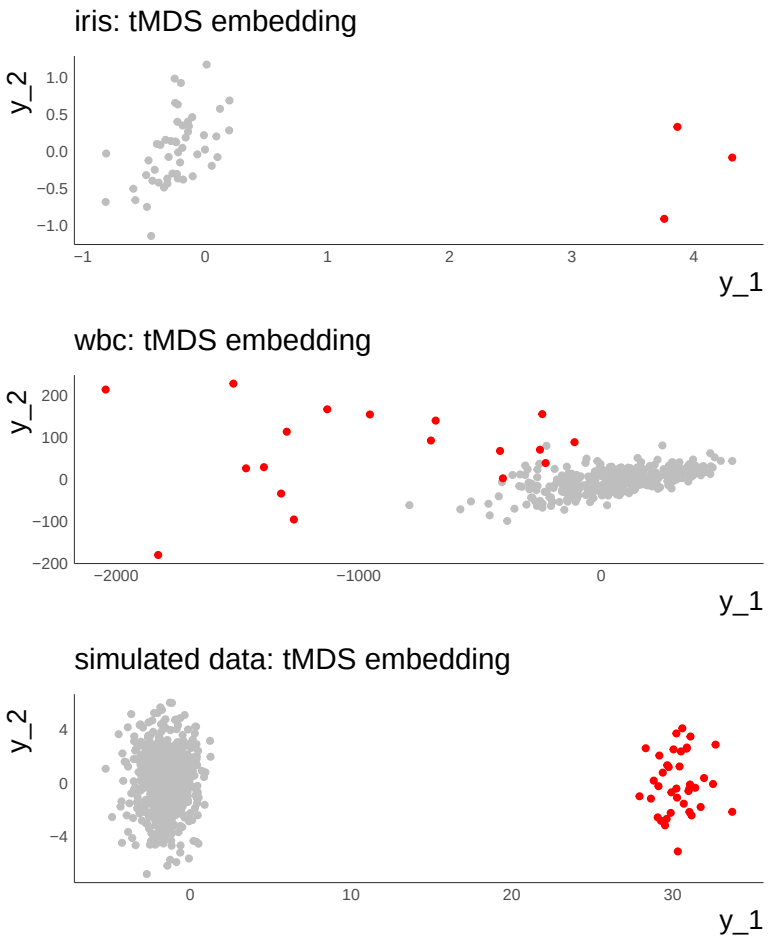

Figure 6: Plots to Table 1 B: tMDS embeddings of tabular data. Inlier class in grey, outlier class in red. $r = 0.05$

10. Enhancing Cluster Analysis via Topological Manifold Learning

Chapter 10 focuses on cluster analysis. It is demonstrated that leveraging the topological structure of data sets improves cluster detection. The manifold learning method UMAP is used to infer the connected components of a data set and the resulting embedding vectors are then used as inputs for the density-based clustering methods DBSCAN. Based on theoretical arguments and extensive qualitative and quantitative experiments the advantages and caveats of this approach are evaluated. As a by-product, the results are compared to the results of other clustering approaches presented in the literature.

Contributing article:

Herrmann, M., Kazempour, D., Scheipl, F., & Kröger, P. (2022). Enhancing cluster analysis via topological manifold learning. arXiv preprint arXiv:2207.00510. http://arxiv.org/abs/2207.00510

Copyright information:

Author contributions:

Moritz Herrmann and Daniyal Kazempour jointly developed the idea of dealing with the topic in this way. Daniyal Kazempour wrote the background part on cluster analysis in Section 2.1 and the part on DBSCAN in Sections 2.3 and 3.2. Moritz Herrmann wrote the rest of the manuscript, developed the methodology, and conducted the experiments and the formal analysis. Fabian Scheipl and Peer Kröger contributed by proofreading the manuscript.

Supplementary material available at:

Code and data: https://github.com/HerrMo/topoclust

Enhancing cluster analysis via topological manifold learning

Moritz Herrmann[1*], Daniyal Kazempour[2], Fabian Scheipl[1†] and Peer Kröger[2†]

[1]Department of Statistics, Ludwig-Maximilians-Universitt Mnchen, Ludwigstr. 33, Munich, 80539, Bavaria, Germany.
[2]Department of Computer Science, Christian-Albrechts-Universitt zu Kiel, Christian-Albrechts-Platz 4, Kiel, 24098, Schleswig-Holstein, Germany.

*Corresponding author(s). E-mail(s): moritz.herrmann@stat.uni-muenchen.de;
Contributing authors: dka@informatik.uni-kiel.de; fabian.scheipl@stat.uni-muenchen.de; pkr@informatik.uni-kiel.de;
†These authors contributed equally to this work.

Abstract

We discuss topological aspects of cluster analysis and show that inferring the topological structure of a dataset before clustering it can considerably enhance cluster detection: theoretical arguments and empirical evidence show that clustering embedding vectors, representing the structure of a data manifold instead of the observed feature vectors themselves, is highly beneficial. To demonstrate, we combine manifold learning method UMAP for inferring the topological structure with density-based clustering method DBSCAN. Synthetic and real data results show that this both simplifies and improves clustering in a diverse set of low- and high-dimensional problems including clusters of varying density and/or entangled shapes. Our approach simplifies clustering because topological pre-processing consistently reduces parameter sensitivity of DBSCAN. Clustering the resulting embeddings with DBSCAN can then even outperform complex methods such as SPECTACL and ClusterGAN. Finally, our investigation suggests that the crucial issue in clustering does not appear to be the nominal dimension of the data or how many irrelevant features it contains, but rather how *separable* the clusters are in the ambient observation space they

are embedded in, which is usually the (high-dimensional) Euclidean space defined by the features of the data. Our approach is successful because we perform the cluster analysis after projecting the data into a more suitable space that is optimized for separability, in some sense.

Keywords: Cluster analysis, Manifold learning, Topological data analysis

1 Introduction

Clustering is the task of uniting similar and separating dissimilar observations in a dataset (Kriegel et al, 2009; Aggarwal, 2014). It is a fundamental task in data analysis and is thus widely investigated in many fields. With this study, we intend to raise awareness for topological aspects of clustering and to provide empirical evidence that topologically-informed approaches which are conceptually and computationally simple can compete with or even outperform much more complex existing methods on a wide range of problems.

1.1 Problem specification

Cluster analysis is usually approached in an algorithm-driven manner, and considerations about the underlying principles of data generating processes and data structures are often limited to a probabilistic conceptualization assuming that the data X follow a joint probability distribution $P(X)$ (Hastie et al, 2009) or, more precisely, a mixture of distributions (Aggarwal, 2014). In contrast, connections to topological data analysis (TDA) (Chazal and Michel, 2021; Wasserman, 2018), a branch of statistical data analysis inferring the structure of data leveraging topological concepts, are usually not considered. In general, the topological aspects of cluster analysis appear to be an under-investigated topic. Current textbooks on cluster analysis (Aggarwal and Reddy, 2014; Aggarwal, 2015; Giordani et al, 2020; Scitovski et al, 2021; Hennig et al, 2015, e.g) and recent reviews of the field (Jain et al, 1999; Kriegel et al, 2009; Assent, 2012; Pandove et al, 2018; Mittal et al, 2019, e.g.) rarely mention the term "topology".

Following Niyogi et al (2011), we consider clustering a natural example of TDA. Since an improved understanding of the underlying principles governing the problem is likely to lead to more suitable methods and novel solutions, our work aims to reduce this lack of awareness of topological aspects in the clustering literature. Specifically, our approach follows Niyogi et al (2011, p. 2) who state that "clustering is a kind of topological question" which tries to separate the data into "connected components." One particularly relevant consequence of this topological perspective is its implication that the difficulty of a clustering problem is not necessarily determined by the data's (nominal) dimensionality.

1.2 Scope of the study

In this work, we make use of the well-known algorithm DBSCAN (Ester et al, 1996) for cluster detection and the recently developed manifold learning algorithm UMAP (McInnes et al, 2018) to infer the topological structure of a dataset.

UMAP has a decidedly topological underpinning, so it is suitable for a theoretical analysis from the clustering perspective we take here. In particular, it builds on simplicial complexes to obtain a fuzzy topological representation of the inherent structure of a dataset. As such, it is based on the same theoretical principles as topological data analysis (Chazal and Michel, 2021; Wasserman, 2018). In addition, it has already been shown that preprocessing by UMAP can improve clustering results (Allaoui et al, 2020) and that the resulting embeddings frequently yield "more compact clusters than t-SNE [another state-of-the-art manifold learning method] with more white space in between" (Kobak and Linderman, 2021, p. 157).

To be specific, "inferring the topological structure" as we do here with UMAP has two aspects: first, a fuzzy graph representation of the dataset is used to find the (number of) connected components. Second, this structure is represented by embedding vectors (i.e. coordinates in a representation space) that are optimized for the separability of the connected components. As we show in section 3, UMAP's graph construction and graph embedding steps both increase cluster separability, and their combined effect thus improves clusterability dramatically.

DBSCAN, on the other hand, is a widely used and well-established method for cluster detection (Schubert et al, 2017). In particular, it neither requires a pre-specified number of clusters nor does it make any assumptions about their specific shapes or patterns. This is important, as inferring the connected components of a dataset is largely equivalent to identifying the clusters it contains. Moreover, the optimized representation of the topological approach focuses on the separability of clusters, not on the specific shapes the clusters might have. Also note that UMAP's developers conjectured that it might enhance density-based clustering, but that this requires further investigation (McInnes, 2018).

From a practical perspective, this means we use UMAP to preprocess the data such that its representation is optimized for separability and use the resulting embedding vectors as inputs for DBSCAN. Although the theoretical and empirical considerations outlined above show that these two methods are suitable, it has to be emphasized that this does not mean that we consider UMAP and DBSCAN the most suitable combination in general. Certainly, additional research has to focus on the pros, cons, and differences between UMAP and other manifold learning methods, in particular t-SNE, and some efforts have already been made in this direction (Kobak and Linderman, 2021; Wang et al, 2021). In this paper, we intend to show that a topological perspective, in general, can improve understanding and practical feasibility of clustering and not whether that specific combination of methods is the most

suitable. Other combinations of clustering and/or manifold learning methods than UMAP and DBSCAN are possible and certainly deserve investigation as well.

Moreover, note that there are other approaches to infer the topological structure of a dataset. For example, persistent homology – which also builds on simplicial complexes – quantifies the topological structure of a dataset by providing information on statistically significant persistent topological features such as connected components, holes, or voids, e.g. (Wasserman, 2018). In contrast, measures of data separability such as the distance-based separability index (Guan and Loew, 2021) quantify the separability of datasets in a single scalar value. However, both approaches only contribute to the first aspect of inferring the topological structure, i.e. they do not provide data representation optimized for separability.

1.3 Contributions

This study makes three distinct contributions: First, section 3 illustrates that approaches motivated by a topological perspective can dramatically reduce the complexity of clustering for both low- and high-dimensional data. This is achieved with an in-depth analysis of simulated data specifically designed to reflect some often described problems of clustering including high-dimensional data, clusters of different densities, and irrelevant features. In addition, a simple toy example demonstrates why and how inferring the intrinsic topological structure of a dataset with UMAP before clustering improves the clustering performance of DBSCAN.

Secondly, with intuition and motivation in place, section 4 is devoted to specific implications of the topological perspective. We describe which structures of a dataset are preserved when inferring the topological structure by finding connected components and enhancing separability (using the UMAP algorithm), in particular by contrasting topological against geometrical characteristics in a detailed qualitative and quantitative analysis of simple synthetic examples.

Finally, in section 5, we report extensive experiments using real-world data. Our results show that inferring the topological structure of datasets before clustering them not only improves – dramatically, for some examples such as MNIST – performance of DBSCAN, but also drastically reduces its parameter sensitivity. The comparatively simple approach of combining UMAP and DBSCAN can even outperform recently proposed clustering methods such as ClusterGAN (Mukherjee et al, 2019), which require expensive hyperparameter tuning, on complex datasets.

In addition, related work and the methods used are described in section 2, while the results are discussed in section 6 before we conclude in section 7.

2 Methods and related work

In this section, we first describe the background of the study and related work, before we outline the methods DBSCAN and UMAP, which are used for clustering and inferring topological structure, respectively, in this study. Readers which are familiar with the methods might skip the corresponding paragraphs. However, note that we will refer to some of the more technical details outlined here in section 3.2.

2.1 Background and related work

The body of literature on clustering, topological data analysis, and manifold learning is extensive and has seen contributions from many different areas and perspectives. General reviews on clustering have been provided for example by Jain et al (1999) and more recently by Saxena et al (2017). Moreover, there a several reviews focusing on cluster analysis for high-dimensional data (Kriegel et al, 2009; Assent, 2012; Pandove et al, 2018; Mittal et al, 2019). In addition, there exist overviews on TDA (Niyogi et al, 2011; Chazal and Michel, 2021; Wasserman, 2018, e.g.) as well as on manifold and representation learning (Cayton, 2005; Bengio et al, 2013; Wang et al, 2021) including the textbooks by Ma and Fu (2012) and Lee and Verleysen (2007).

The variety of clustering algorithms is vast and endeavors have been made to capture this diversity through taxonomies. DBSCAN, the algorithm used here, is a density-based approach. One of its major advantages is that it does not require a pre-specified number of clusters and that the clusters can have arbitrary shapes and patterns. Its hierarchical version (HDBSCAN, Campello et al, 2013) does not use a global ε-threshold but computes on its own multiple cut-off values resulting in clusters of different densities and therefore requires only the $minPts$ parameter. Similar to HDBSCAN, the OPTICS algorithm (Ankerst et al, 1999) calculates an ordering of the observations without a global ε-threshold that provides broader insight on the structure of the data. However, the method does not explicitly assign cluster memberships. Instead, it allows viualizing the hierarchical cluster structure for example via reachability plots (Ankerst et al, 1999).

Further categories are *hierarchical* and *partitioning* algorithms (Jain et al, 1999), where the latter can be divided further into sub-taxonomies. Some of them are based on the minimization of distances to certain prototypes (centroids, medoids, etc.), this includes algorithms like k-means (Lloyd, 1982), or its more general archetype of algorithms: Gaussian Mixture Models (GMMs) among which the Expectation-Maximization (EM) algorithm (Dempster et al, 1977) is a prominent exponent. A major caveat, however, is that these methods estimate a specific probabilistic model which includes the number of clusters to be detected and often fail if the data is distributed differently (Liu and Han, 2014).

In contrast, *spectral* clustering, a family of algorithms that shares some common ground with many manifold learning methods that are also based

on spectral decompositions of pairwise (dis)similarity matrices, is more robust with respect to the shape and distribution of the clusters. However, these methods require the number of clusters to be specified in advance (Von Luxburg, 2007; Liu and Han, 2014).

Subspace clustering approaches emerged specifically for high-dimensional settings (Kriegel et al, 2009; Assent, 2012; Pandove et al, 2018; Mittal et al, 2019). The fundamental assumption here is that objects within a cluster do not exhibit high similarities among all dimensions but only within a small subset of features that can either (a) span an *axis-parallel* subspace or (b) an affine projection to an *arbitrarily-oriented* subspace ("correlation clustering"). In both cases, the objects of a cluster are assumed to be located on a common, low-dimensional linear manifold.

In contrast, manifold learning is based on the assumption that data observed in a high-dimensional ambient observation space is distributed on or near a potentially nonlinear manifold with a much smaller intrinsic dimension than the ambient space (Ma and Fu, 2012). In general, the aim is to find low-dimensional representations of datasets preserving as much of the structure of the observed data as possible. A synonymous term is nonlinear dimension reduction (NDR) (Lee and Verleysen, 2007). However, there is no general definition of which characteristics are to be preserved and represented and different methods infer the intrinsic structure and provide low-dimensional representations in different ways.

For instance, principal component analysis (PCA) yields embedding vectors that optimally preserve global Euclidean distances in the original data space, while other methods such as Isomap (Tenenbaum et al, 2000) yield embedding vectors that aim to preserve geodesic distances on a single, globally connected data manifold. Methods like t-distributed Stochastic Neighbor Embedding (t-SNE, van der Maaten and Hinton, 2008) and uniform manifold approximation and projection (UMAP, McInnes et al, 2018) have been successfully applied to complex high-dimensional datasets with cluster structure. More recently, methods with a specific topological focus such as general purpose Topomap (Doraiswamy et al, 2021) as well as domain specific Paga (Wolf et al, 2019), which focuses on the analysis of single cell data, have been proposed. The manifold learning-based clustering approach of Souvenir and Pless (2005) relies on the assumption that data is sampled from multiple *intersecting* lower-dimensional manifolds.

Several studies that precede ours also focus on the combination of manifold learning techniques and cluster analysis, with applications to cytometry data (Putri et al, 2019), brain tumor segmentation (Kaya et al, 2017), spectral clustering (Arias-Castro et al, 2017), or big data (Feldman et al, 2020), the latter three based on PCA. DBSCAN was used in combination with multi-dimensional-scaling (MDS) in Mu et al (2020), and UMAP was used for time-series clustering (Pealat et al, 2021) as well as clustering SARS-COV-2 mutation datasets (Hozumi et al, 2021). However, these all focus on specific domains and not on the underlying topological principles. In contrast, we base

our work on a topological perspective on clustering first described theoretically by Niyogi et al (2011), who conceptualize clustering as the problem of identifying the *connected components* of a data manifold. We show the theoretical and practical utility of this perspective by means of extensive experiments based on synthetic and real datasets. Similar in spirit to our work, Allaoui et al (2020) perform a comparative study with real data to show that UMAP can considerably improve the performance of clustering algorithms. Among other things, they combined UMAP with HDBSCAN and report comparable clustering results for three of the real-world datasets (Pendigits, MNIST and FMNIST) also used here. However, in contrast to our study, Allaoui et al (2020) do not provide insights into the conceptual topological underpinnings, nor do they describe how the data structures preserved in UMAP embeddings lead to these performance improvements. Note that their results also show empirically that the benefits of the proposed approach are not tied to any particular combination of NDR and clustering methods.

2.2 UMAP

The principle idea behind UMAP essentially consists of two steps:
1) Constructing a weighted k-nearest neighbor (k-NN) graph from a pairwise distance matrix.
2) Finding a (low-dimensional) representation of the graph which preserves as much of its structure as possible.
Note that this is the fundamental principle in manifold learning and the details of the two steps constitute the differences between manifold learning methods (Wang et al, 2021). However, unlike many other manifold learning methods, UMAP is based on a solid theoretical foundation that ensures that the topology of the manifold is faithfully approximated by its fuzzy simplical set representation. We concentrate on the computational aspects outlined in McInnes et al (2018) and refer interested readers to the original study for theoretical details.

2.2.1 Graph construction

Given a dataset $X = \{x_1, ..., x_{n_{\text{obs}}}\}$ sampled from a space equipped with a distance metric $d(x_i, x_j)$, UMAP constructs a directed k-NN graph $\bar{G} = (V, E, w)$ with the vertices V_i being observations x_i from X, E the edges and w the weights, based on the following definitions.

Definition 1 The distance ρ_i of an observation x_i to its nearest neighbor x_{i_j} is defined by

$$\rho_i = \min\{d(x_i, x_{i_j}) | 1 \leq j \leq k, d(x_i, x_{i_j}) > 0\}.$$

Definition 2 A (smooth) normalization factor σ_i is set for each x_i by

$$\sum_{j=1}^{k} \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right) = \log_2(k).$$

This defines a local (Riemannian) metric at point x_i.

Definition 3 Weight function: The edge weights of the graph are defined by

$$w((x_i, x_{i_j})) = \exp\left(\frac{-\max(0, d(x_i, x_{i_j}) - \rho_i)}{\sigma_i}\right).$$

Note, the distance to the nearest neighbor ρ_i ensures that x_i is connected to at least one other point with an edge of weight 1 (local connectivity constraint).

For the theory to work it is essential to assume that the data is uniformly distributed on the manifold, which is too strong an assumption for real-world data. The issue is bypassed by defining independent notions of distance at each observed point through σ_i and ρ_i. However, these local metrics may not be mutually interchangeable, which means that the "distance" between neighboring points x_i and x_j may not be the same if measured w.r.t x_i or w.r.t. x_j, i.e., $d(x_i, x_j) \neq d(x_i, x_j)$, so edge weights in $\bar{G}$ depend on the direction of the edges.

A unified, undirected graph G with adjacency matrix B is obtained by

$$B = A + A^T - A \circ A^T, \tag{1}$$

with A the weighted adjacency matrix of $\bar{G}$ and $\circ$ the point-wise product. Note that Eq. (1) represents the well-defined operation of unioning fuzzy simplicial sets (with which the manifold is approximated). The resulting entries in B can be interpreted as the probability that at least one of the two directed edges between two vertices in $\bar{G}$ exists, or more generally as a measure of similarity between two observations x_i and x_j. Note that it has recently been shown that a stricter notion of connectivity induced by mutual nearest neighbors can further improve the topology preserving property of standard UMAP used here (Dalmia and Sia, 2021).

2.2.2 Graph embedding

The objective is to find a configuration of points in the representation space Y whose fuzzy simplicial set is as similar as possible to the fuzzy simplicial set of the original data, as represented by G. To find this low-dimensional representation, UMAP optimizes the cross entropy of edge weights in the two spaces. Similarities in the observation space are represented in terms of the local smooth nearest neighbor distances as

$$v_{ij} = (v_{j|i} + v_{i|j}) - v_{j|i} v_{i|j}, \tag{2}$$

with $v_{j|i} = \exp[(-d(x_i, x_j) - \rho_i)/\sigma_i]$ (c.f. Eq. (1)), and similarities in the representation space Y as

$$w_{ij} = (1 + a||y_i - y_j||_2^{2b})^{-1}, \tag{3}$$

the cross entropy between the two fuzzy simplicial set representations

$$C_{UMAP} = \sum_{i \neq j} v_{ij} \log\left(\frac{v_{ij}}{w_{ij}}\right) + (1 - v_{ij}) \log\left(\frac{1 - v_{ij}}{1 - w_{ij}}\right) \tag{4}$$

is minimized via stochastic gradient descent (SGD) to obtain the graph layout (by default $a \approx 1.929$ and $b \approx 0.7915$). The two terms in Eq. (4) represent the attractive and repulsive forces for the graph layout algorithm used here. Next to a and b, UMAP's central tuning parameters are the number of nearest neighbors k (often denoted as n or `n_neighbors`), the number of SGD optimisation iterations `n-epochs`, the dimension d of the representation space, and `min-dist`, a parameter controlling how close neighboring points can appear in the representation.

2.3 DBSCAN

The principle idea behind DBSCAN is captured within 6 definitions we adapt from Ester et al (1996) and elaborate on:

Definition 4 ε-neighborhood of an object: The ε-neighborhood of an object x_i denoted by $\mathcal{N}_\varepsilon(x_i)$, is defined by:

$$\mathcal{N}_\varepsilon(x_i) = \{x_j \in X | d(x_i, x_j) \leq \varepsilon\}$$

where X denotes a given dataset.

Definition 5 Directly density-reachable: An object x_i is direct density-reachable from an object x_j w.r.t. a given ε-range and $MinPts$ if:
1) $x_i \in \mathcal{N}_\varepsilon(x_j)$ and
2) $|\mathcal{N}_\varepsilon(x_j)| \geq MinPts$ (core point condition)

Definition 6 Density-reachable: An object x_i is density-reachable from another object x_j w.r.t. ε and $MinPts$ if there is a chain of objects $x_1, ..., x_c, x_1 = x_i, x_c = x_j$ such that x_{l+1} is directly density-reachable from x_l.

Definition 7 Density-connected: An object x_i is density-connected to another object x_j w.r.t. ε and $MinPts$ if there is an object o such that both, x_i and x_j are density-reachable from o w.r.t. ε and $MinPts$.

Definition 8 Cluster: Let X be a given dataset of objects. A cluster C w.r.t. ε and $MinPts$ is a non-empty subset of X satisfying the following conditions:
1) $\forall x_i, x_j$: if $x_i \in C$ and x_j is density-reachable from x_i w.r.t. ε and $MinPts$, then $x_j \in C$ (Maximality)
2) $\forall x_i, x_j \in C : x_i$ is density-connected to x_j w.r.t. ε and $MinPts$ (Connectivity)

Definition 9 Noise: Let $C_1, ..., C_{n_c}$ be the n_c clusters of the given dataset X w.r.t. parameters ε_i and $MinPts_i$, $i = 1, ..., n_c$. Then noise is defined as the set of objects in the dataset X that do not belong to any cluster C_i, i.e. $noise = \{x_i \in X | \forall i : x_i \notin C_i\}$

In Definition 5 an object is a core point if it has at least $MinPts$ number of objects within its ε-neighborhood. In the case that no objects in a given dataset are density-reachable then we would obtain n_c clusters where n_c denotes the number of core-points in a dataset X for a given ε and $MinPts$. This means that the number of core points can be considered as an upper bound for the number of emerging clusters for a given ε and $MinPts$. Further it can be deduced from the core point definition that the region surrounding a core point is *more dense* compared to density-connected objects that do not satisfy $|\mathcal{N}_\varepsilon(x_j)| \geq MinPts$ meaning that they are objects in more *spare* regions.

3 Inferring the topological structure enhances clusterability

In this section, we demonstrate that the correct use of manifold learning (here, specifically: UMAP), as motivated by our topological framing, largely avoids several frequently described challenges in cluster analysis.

A major problem affecting cluster analysis is that clustering often becomes more challenging in high-dimensional datasets. Specifically, the presence of many irrelevant and/or dependent features potentially degrades results (Kriegel et al, 2009). However, contrary to widespread "folk-methodological" superstitions and some sources like Assent (2012), the well-known result that L_p distances lose their discriminating power in high dimensions (Beyer et al, 1999, e.g.) is entirely irrelevant for well-posed clustering problems: both the original publication and subsequent works like Kriegel et al (2009) and Zimek and Vreeken (2015) show that the conditions for this result do not apply if the data is distributed in well separable clusters. In particular, this means that DBSCAN, being based on pairwise distance information, can easily detect clusters in high-dimensional datasets.

Nevertheless, there are other problems specific to density-based clustering, and DBSCAN in particular, among which finding a suitable density level is one of the most important (Kriegel et al, 2011; Assent, 2012). A recent review (Schubert et al, 2017), outlined some heuristic rules for specifying ε for DBSCAN, but domain knowledge should mostly determine such decisions. More importantly, density-based clustering is likely to fail for clusters with varying densities. In such cases, a single global density level – for example, specified via ε in DBSCAN – cannot delineate cluster boundaries successfully (Kriegel et al, 2011).

In addition to these well-known issues, we outline another more subtle, less well-known aspect: not only does the difficulty of a clustering problem not necessarily increase for high-dimensional X, but clusters may even become easier to detect in higher dimensional (embedding) spaces.

3.1 Enhancing clusterability of DBSCAN with UMAP

The four example datasets we consider here illustrate the following three points: (1) Density-based clustering works in some but not all high-dimensional

settings. (2) Perfect performance may not be achievable even for extensive parameter grid searches, and suitable ε values are highly problem-specific. (3) Most importantly, manifold learning can considerably enhance clustering both by improving performance and by reducing parameter sensitivity of DBSCAN to the extent that it becomes almost tuning-free.

The datasets we consider here consist of three clusters sampled from three multivariate Gaussian distributions with different mean vectors. In the first two examples, denoted by E_{100} and E_{1000}, the covariance matrix for all three Gaussians is the identity matrix, inducing clusters of similar densities. In the latter two examples, U_3 and U_{1003}, the covariance matrices differ, inducing clusters of different density. In addition, we consider problems with very different dimensionalities. Observations in setting E_{100} are sampled from 100-dimensional Gaussians, while observations in setting E_{1000} are sampled from 1000-dimensional Gaussians. In contrast, observations for U_3 and U_{1003} are sampled from 3-dimensional Gaussians. For U_{1003}, an additional 1000 features that are irrelevant for cluster membership are sampled independently and uniformly from $[0, 1]$. For each setting, we sample 500 observations from each of the three clusters, i.e. each example dataset consists of 1500 observations in total. The complete specifications of the examples are given in Table 1.

Table 1 Specifications of the settings E_{100}, E_{1000}, U_3, and U_{1003}. In setting U_{1003} clusters are defined by means of $p = 3$ dimensional Gaussians, yet an additional 1000 irrelevant features are sampled uniformly from $[0, 1]$, leading to a total dimensionality of 1003.

Setting	p	Means	Variances
E_{100}	100	$\boldsymbol{\mu}_i \in \{\mathbf{0, 0.5, 1}\}$	$\sigma_i = 1$
E_{1000}	1000	$\boldsymbol{\mu}_i \in \{\mathbf{0, 0.5, 1}\}$	$\sigma_i = 1$
U_3	3	$\boldsymbol{\mu}_i \in \{\mathbf{0, 3, 7}\}$	$\sigma_i \in \{0.1, 1, 3\}$
U_{1003}	3	$\boldsymbol{\mu}_i \in \{\mathbf{0, 3, 7}\}$	$\sigma_i \in \{0.1, 1, 3\}$

Figure 1 shows the Adjusted Rand Index (ARI) (Hubert and Arabie, 1985, Eq. 5) and the Normalized Mutual Information (NMI) with maximum normalization (Vinh et al, 2010, Tab. 2) for different ε values obtained by either applying DBSCAN directly to the observed data or to their 2D UMAP embeddings. Both measures compare two data partitions and return a numeric value quantifying the agreement. While the NMI strictly ranges between $[0, 1]$ (with a value of 1 indicating perfect concordance), the ARI is 0 only if the Rand Index exactly matches its expected value under the null hypothesis that the partitions are generated randomly from a hypergeometric distribution (Hubert and Arabie, 1985).

Several aspects need to be emphasized. First of all, the effect of the dimensionality of the dataset on the performance of DBSCAN applied to the original data is complicated (Figure 1, first column (A)). Contrary to preconceived notions, it can be easier to detect clusters in higher dimensions. Figure 1 A shows that using only DBSCAN, clusters are more easily detected in the

1000-dimensional data (2nd row) than in the 100-dimensional data (1st row, although perfect performance is not achieved by DBSCAN in either of the two.

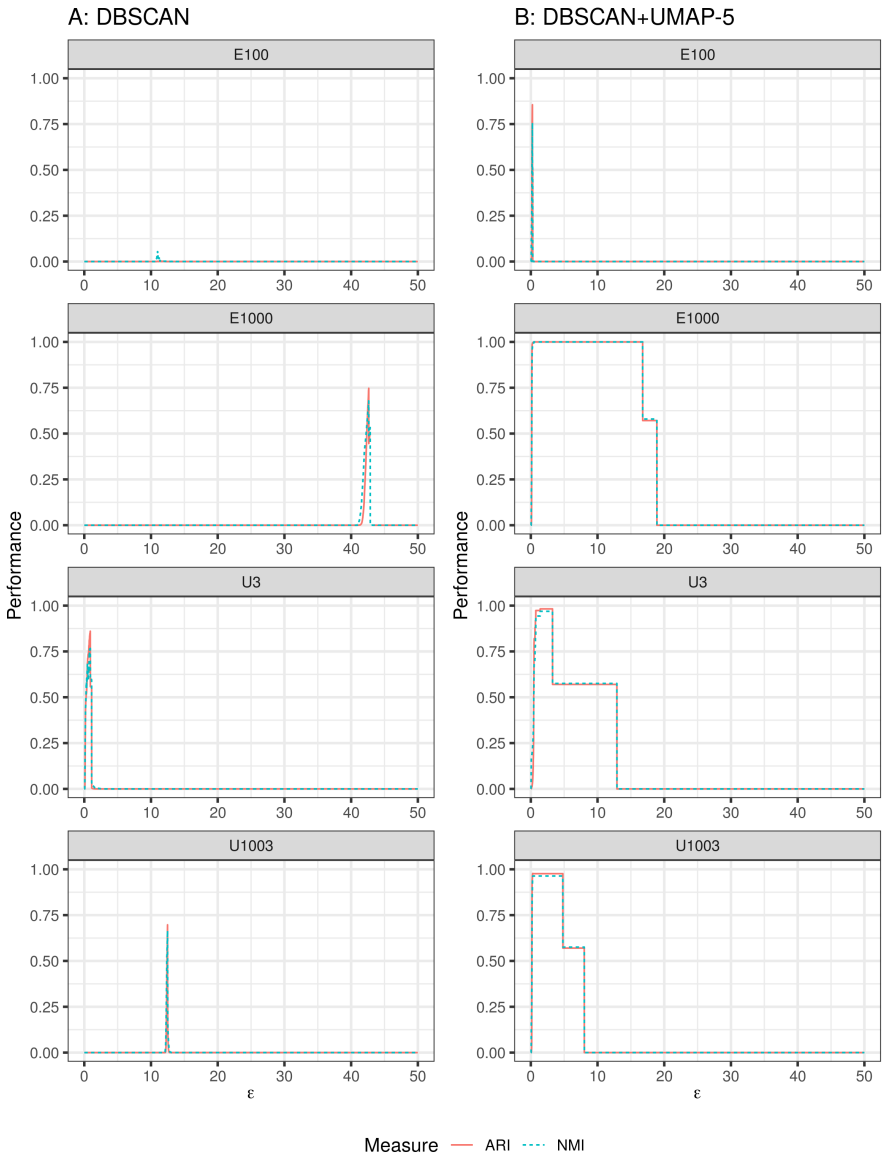

Fig. 1 ARI and NMI as a function of ε for the synthetic settings $E_{100}, E_{1000}, U_3, U_{1003}$. First column: DBSCAN directly applied to the data. Second column: DBSCAN applied to a 2D UMAP embedding with $k = 5$. Clusters sampled from multivariate Gaussian distributions (see Table 1 for specifications). For setting U_{1003}, additional 1000 irrelevant variables are sampled uniformly from $[0, 1]$. DBSCAN computed for $\varepsilon \in [0.01, 50]$, step size: 0.01; $minPts = 5$.

The dimension of the Gaussian distributions defining the clusters is the only difference between these two settings. On the other hand, Figure 1 A, shows that it can also be the other way round. In the 1003-dimensional dataset with 1000 irrelevant features (4th row), cluster performance is much lower than in the corresponding 3-dimensional dataset with only 3 relevant variables (3rd row). Again, perfect cluster performance is not achieved by DBSCAN alone. Note that settings U_3 and U_{1003} define clusters with varying densities, so DBSCAN is expected not to provide a perfect result.

Secondly, finding a suitable value of ε is very challenging using DBSCAN alone. Note that the optimal ε_{opt} varies between 0.9 and 42.64 for these examples. Identifying a suitable ε is even more problematic since the sensible ε-ranges are very small (e.g. see U_{1003}). In some cases, clustering does not seem feasible at all even with an optimally chosen ε – optimal results are very poor for setting E_{100} with ARI (NMI) $= 0.003(0.05)$ for $\varepsilon_{opt} = 11.32(10.98)$. Moreover, while ε_{opt} is not necessarily consistent for datasets with approximately the same dimensionality – compare $\varepsilon_{opt} = 42.64$ for E_{1000} to $\varepsilon_{opt} = 12.48$ for U_{1003} – it can be similar for datasets with very different dimensionality – compare $\varepsilon_{opt} \sim 11$ for E_{100} to $\varepsilon_{opt} = 12.48$ for U_{1003}.

Finally, the crucial point we want to highlight with these examples is that inferring the topological structure before clustering by applying DBSCAN on UMAP embeddings instead of directly to the data makes all these issues (almost) completely disappear (see Figure 1 B). First of all, clustering performance is increased in all four examples; in three it even leads to perfect performances. But not only is performance increased, but UMAP also dramatically reduces the complexity of finding a suitable ε. In all considered cases the sensible ε-ranges start near zero, rapidly reach the optimal value, and remains optimal over a wide range of ε-values in three of the four examples. Note that we do not tune UMAP at all – we simply set $k = 5$ and leave all other settings at their default values.

We emphasize that perfect performance is obtained for large swaths of the ε-range we consider for the two high-dimensional examples. This suggests that the crucial issue in clustering is not the nominal dimension of the dataset or whether it contains irrelevant features, but rather how separable the clusters are in their ambient space, which is usually simply the p-dimensional Euclidean space spanned/defined by the dimensions/features of the data, while the approach taken here attempts to cluster observations after projecting them into a space that is optimized for separability.

In summary, applying DBSCAN on UMAP embeddings not only improved performance considerably, but it also reduced the sensitivity of DBSCAN w.r.t. ε. In particular, suitable ε-ranges started near zero for all considered examples. Our experiments described in section 5 show that this holds for complex real data such as fashion MNIST (Xiao et al, 2017) as well, where applying DBSCAN on UMAP embeddings not only dramatically improved

DBSCAN's performance but even outperformed the recently proposed ClusterGAN (Mukherjee et al, 2019) method. In the next subsection, we examine the technical aspects that explain this behavior in a simple toy example.

3.2 Reasons for improved clusterability

This section lays out possible reasons for the observed improvements w.r.t clusterability with a detailed analysis of the underlying technical mechanisms in a simple toy example. Consider the following distance matrix between six objects:

$$\begin{pmatrix} 0 & 0.6 & 0.7 & 1.3 & 1.2 & 1.5 \\ 0.6 & 0 & 0.5 & 0.75 & 1.6 & 1.3 \\ 0.7 & 0.5 & 0 & 1.4 & 1.3 & 1.1 \\ 1.3 & 0.75 & 1.4 & 0 & 0.7 & 0.75 \\ 1.2 & 1.6 & 1.3 & 0.7 & 0 & 0.75 \\ 1.5 & 1.3 & 1.1 & 0.75 & 0.75 & 0 \end{pmatrix} \tag{5}$$

Inspecting this distance matrix reveals two clusters of objects, shown here in green and cyan. We set DBSCAN's core point condition parameter to $minPts = 2$. Note that the object itself is not considered part of its ϵ-neighborhood. We set $\varepsilon = 0.75$, so that every object whose row (or column) in the distance matrix contains at least two entries ≤ 0.75 is considered a "core point". Since two objects from the different clusters have a distance of exactly 0.75 (orange entries), all objects are part of a single *connected component*, and the two dense regions are subsumed into a single large cluster for $\varepsilon = 0.75$, as can be seen in the matrix below:

$$\begin{pmatrix} 0 & 0.6 & 0.7 & 1.3 & 1.2 & 1.5 \\ 0.6 & 0 & 0.5 & 0.75 & 1.6 & 1.3 \\ 0.7 & 0.5 & 0 & 1.4 & 1.3 & 1.1 \\ 1.3 & 0.75 & 1.4 & 0 & 0.7 & 0.75 \\ 1.2 & 1.6 & 1.3 & 0.7 & 0 & 0.75 \\ 1.5 & 1.3 & 1.1 & 0.75 & 0.75 & 0 \end{pmatrix} \tag{6}$$

To avoid this collapsed solution, one could try to reduce the ε parameter to e.g. $\varepsilon = 0.74$. However, as a consequence, now all the objects in the second (cyan) cluster become "noise": They no longer satisfy the "core point" condition for $minPts = 2$, since at most one distance in each of their rows is ≤ 0.74. This means only one cluster (top left, green) is detected, as can be seen in the following matrix:

$$\begin{pmatrix} 0 & 0.6 & 0.7 & 1.3 & 1.2 & 1.5 \\ 0.6 & 0 & 0.5 & 0.75 & 1.6 & 1.3 \\ 0.7 & 0.5 & 0 & 1.4 & 1.3 & 1.1 \\ 1.3 & 0.75 & 1.4 & 0 & 0.7 & 0.75 \\ 1.2 & 1.6 & 1.3 & 0.7 & 0 & 0.75 \\ 1.5 & 1.3 & 1.1 & 0.75 & 0.75 & 0 \end{pmatrix} \tag{7}$$

From this first example, we conclude 1) that there may be cases where even a single object may *connect* two clusters, yielding a single collapsed cluster and 2) that the sensitivity of clustering solutions to hyperparameter settings is large: A small change of the ε-parameter by only 0.01 led to a fundamentally different solution.

Thus, we should look for improvements that (i) reduce the sensitivity of results towards the parameter settings and (ii) increase the separability of the data and thereby reduce the susceptibility of DBSCAN to merge multiple poorly separated clusters via interconnecting observations at their respective margins. Sharpening the distinction between dense and sparse regions within the dataset, i.e. increasing separability, improves clusterability. As we will now see, UMAP is able to do exactly that by arranging objects into clusters with fairly constant density within and empty regions in between.

To illustrate this, we consider the representation of the toy example via the fuzzy graph as constructed by UMAP. This reflects the fuzzy simplicial set representation of the data and crucially depends on the number of nearest neighbors k. We start with $k = 6$. This leads to a graph with adjacency matrix

$$\begin{pmatrix} 0 & 1.0 & 0.95 & 0.29 & 0.53 & 0.25 \\ 1.0 & 0 & 1.0 & 0.9 & 0.19 & 0.30 \\ 0.95 & 1.0 & 0 & 0.24 & 0.45 & 0.58 \\ 0.29 & 0.9 & 0.24 & 0 & 1.0 & 1.0 \\ 0.53 & 0.19 & 0.45 & 1.0 & 0 & 1.0 \\ 0.25 & 0.3 & 0.58 & 1.0 & 1.0 & 0 \end{pmatrix} \tag{8}$$

Each cell represents the fuzzy edge weight v_{ij} (Eq. 2) connecting two points, so each value represents the affinity of two observations, not their dissimilarity as in the distance matrices before. As before, the cluster structure is obvious in this representation, with high affinities (≥ 0.95) where distances had been low (≤ 0.75). The representation learned by UMAP in the graph construction step clearly reflects the cluster structure of the dataset.

Note that this fuzzy topological representation by itself already amplifies the cluster structure: if we stopped UMAP at this point and converted the affinities v_{ij} into dissimilarities e.g. via $d_{ij} = 1 - v_{ij}$, $i \neq j$, DBSCAN with $minPts = 2$ would yield perfect cluster results for $\varepsilon \in [0.01, 0.09]$!

Note as well that UMAP's graph layout optimization has not even been performed yet and that the nearest-neighbor parameter k has been set to 6, the largest possible value in this example. Thus, the vast improvement in separability we observe is due only to the way UMAP learns and represents the structure of the data in the fuzzy graph G alone. The improvement can be driven even further both by decreasing the parameter k and by conducting the graph layout optimization.

First, consider the effect of k. In the following, blanks in the matrices denote zero entries. Graph 9 shows G for $k = 3$. Clearly, the beneficial effects

we noted for $k = 6$ are considerably amplified.

$$\begin{pmatrix} & 1.0 & 0.83 & & & \\ 1.0 & & 1.0 & 0.58 & & \\ 0.83 & 1.0 & & & & \\ & 0.58 & & & 1.0 & 1.0 \\ & & & 1.0 & & 1.0 \\ & & & 1.0 & 1.0 & \end{pmatrix} \tag{9}$$

Almost all v_{ij} become zero (i.e. there is no affinity/similarity between the two points) except for those joined in one of the clusters and the two entries which caused DBSCAN to break. Turning v_{ij} into d_{ij} as above, DBSCAN yields correct clusters for $\varepsilon \in [0.01, 0.42]$.
By setting $k = 2$, the smallest possible value due to the local connectivity constraint, we can further distill the cluster structure down to its bare essentials:

$$\begin{pmatrix} & 1.0 & & & & \\ 1.0 & & 1.0 & & & \\ & 1.0 & & & & \\ & & & & 1.0 & 1.0 \\ & & & 1.0 & & \\ & & & 1.0 & & \end{pmatrix} \tag{10}$$

Based on this graph, DBSCAN yields correct clusters for $\varepsilon \in [0.01, 0.99]$! Thus, by setting the nearest neighbor parameter of UMAP to a very small value, the cluster separability is dramatically amplified and DBSCAN's sensitivity w.r.t. ε is significantly reduced.
However, the graph layout optimization step has not even been performed yet. This additional step is crucial, in particular for reducing the parameter sensitivity of clustering methods. This is due to the fact $d_{ij} = 1 - v_{ij}$ only converts affinities into dissimilarities. Finding a graph layout via the cross-entropy C_{UMAP} as defined in Eq. 4 instead not only converts affinities (indirectly) into dissimilarities but also improves the conversion itself w.r.t. to separability (on top of the separability gained by the graph construction), since the optimization procedure optimizes the graph layout for increased cluster separability. This can be explained as follows:

C_{UMAP} becomes minimal for $v_{ij} = w_{ij}$. For $v_{ij} = 0$, the further away from each other the embedding vectors y_i and y_j are placed, the better, since this will drive w_{ij} towards zero. Considering graphs 9 and 10, we see that v_{ij} is zero mostly for observations from different clusters. Minimizing C_{UMAP} thus increases cluster separability in the embedding space by driving objects from different clusters apart. Note that minimizing the cross entropy "can be seen as an approximate bound-optimization (or Majorize-Minimize) algorithm [...] implicitly minimizing intra-class distances and maximizing inter-class distances" (Boudiaf et al, 2020, p. 3). The optimization in the graph embedding step of UMAP thus leads to tighter clusters with more white space in between.

The most relevant additional benefit this graph embedding step provides is the large expansion of well-performing ε-ranges for DBSCAN. Since the graph layout optimization uses stochastic gradient descent, the resulting embedding vectors are not deterministic. To account for this randomness, we perform 25 embeddings for each value of k and compute separate averages of the lower and the upper interval boundaries of the ε-ranges yielding optimal cluster performance. On average, the obtained embedding coordinates yield correct clusters for $k = 6$ with $\varepsilon \in [0.83, 1.03]$, for $k = 3$ with $\varepsilon \in [0.70, 6.76]$, and for $k = 2$ with $\varepsilon \in [0.79, 20.94]$. Even the smallest (optimal) ε-ranges we observed over the 3×25 replications are at least as large as the ones obtained on the fuzzy graph for $k = 6$, and still considerably larger for $k = 3$ and $k = 2$: $[0.94, 1.03]$, $[0.72, 1.33]$, $[1.16, 4.57]$, respectively. Further analysis of the variability resulting from optimizing embedding vectors via SGD can be found in appendix A.

These results indicate how crucial optimizing separability by computing embedding vectors is for clustering performance. Appendix B confirms its importance on real data.

In these and the following experiments, all of UMAP's other hyperparameters were left to the implementation defaults, in particular `min_dist` $= 0.1$. Additionally adjusting these parameters might further increase separability. However, tuning parameters in an unsupervised setting is a notoriously difficult task and since the results are already convincing by setting k to a small value, we concentrate on the effect of k.

In summary, both the graph construction and the graph embedding steps in the UMAP algorithm independently contribute to an increased separability of clusters in a dataset, and their combined effect improves clusterability dramatically.

4 The price to pay: structures preserved and lost

As we have outlined in the previous sections, UMAP is able to infer and even enhance the topological, i.e. the cluster, structure of a dataset. However, these improvements come at a price which will be outlined in this section.

4.1 Topology vs. geometry

Beyond topological structure, i.e., mere "connectedness", datasets also have geometrical structure – the shapes of the clusters and how the clusters are positioned relative to each other in the ambient space.

Consider the example of a dataset consisting of three nested spheres embedded in a 3-dimensional (Euclidean) space (see Figure 2 A). What kind of structure does this dataset yield? First of all, from a purely topological perspective, we have three unconnected topological subspaces, i.e. clusters: the three spheres.

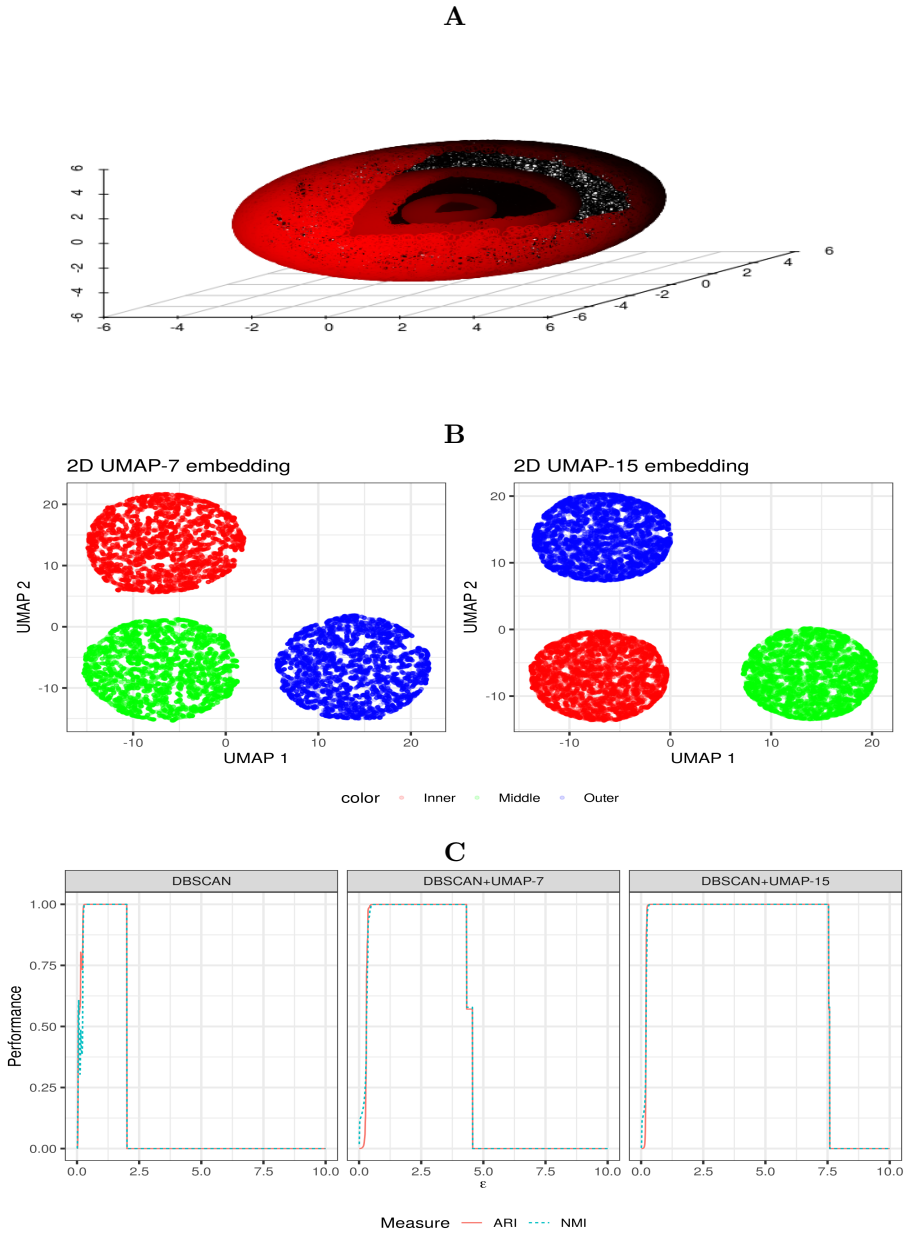

Fig. 2 Effects of UMAP: preservation of topological vs. geometrical structure. **A**: Three nested spheres in 3D ($n_{\text{obs}} = 30000$, part of the data omitted to make the nested structure visible). **B**: UMAP embeddings for $k = 7$ and $k = 15$. The clusters, i.e. topological structure, is preserved. Geometrical structure is not preserved: Ambient space geometry ("nestedness") is lost; for $k = 7$, less of the spherical/circular shape is preserved. **C**: Clustering performances for $\varepsilon \in [0, 10]$ (step size: 0.01, $minPts = 5$) for DBSCAN directly applied to the data (left) and applied to the UMAP embeddings ($k = 7$: middle, $k = 15$: right).

Moreover, from an additional geometrical perspective, we have information on the shape of the individual clusters: they form spheres, i.e. 2-dimensional surfaces. Finally, we have information on the relative position of the clusters to each other within the ambient feature space: the spheres are nested.

What happens if these data are represented in a 2D UMAP embedding? Since a sphere cannot be isometrically mapped to a 2-dimensional plane, some distortion of the geometric structure will be unavoidable in any 2D embedding. Figure 2 B shows that, in fact, most of the geometrical structure is lost in UMAP embeddings: the relative positioning of the clusters diverges from the original data and is not consistent over different embeddings. The effect on the shape of the clusters is less severe. While for $k = 15$ the embeddings are similar to circles, i.e. 2D spheres, for $k = 7$ the general circular shape is retained, yet less uniformly. In contrast, the topological structure of the different clusters is not only preserved in full but even exaggerated – clusters are much more separated in the embeddings, which is also reflected once again in much wider ε-ranges that yield sensible results (Figure 2 C). DBSCAN alone provides perfect clustering performances only over a much smaller ε range than when applied to these UMAP embeddings.

As a further example, we consider the complex 2D synthetic dataset by Jain (2010), "who suggest that it cannot be solved by a clustering algorithm" (Barton et al, 2019, p. 2). This "impossible" data contains seven clusters with complex structure, see Figure 3 A. The clusters have different densities, are in part non-convex, and are not linearly separable. DBSCAN by itself is not able to detect the full cluster structure and choosing ε from $[0, 15]$ (step size: 0.01 $minPts = 5$) based on an optimal ARI value yields a very different cluster result than choosing ε based on the optimal NMI value (see Figure 3 B & C). This challenging example further demonstrates two important points:

First, how successfully UMAP embeddings preserve the connected components (i.e. topological structure) and simultaneously distort geometric structure. In Figure 3 D, we can see that the nested structure of the circles and the entanglement of the spirals are completely lost and that the spirals have been "unrolled" in the embedding space, but the different clusters are very clearly separated.

Second, the example illustrates that "dimension inflation" via UMAP can have a positive effect on cluster performance. "Dimension inflation" means that the data is embedded into a space of higher dimensionality than the observed data. Although this is uncommon and we are not aware of any work where this has been investigated before, there are no restrictions that prevent UMAP from being used in this way. Consider Figure 3 F, which shows ARI- and NMI-curves obtained with DBSCAN applied (1) to the data, (2) a 2D UMAP-5, and (3) a 3D UMAP-5 embedding. Although the 2D UMAP-5 embedding already improves performance and strongly reduces parameter sensitivity, it does not yield a perfect solution. In the 2D embedding (Fig. 3 D), the two spirals are very close to each other, with a gap between them that is smaller than the gap appearing within the black cluster.

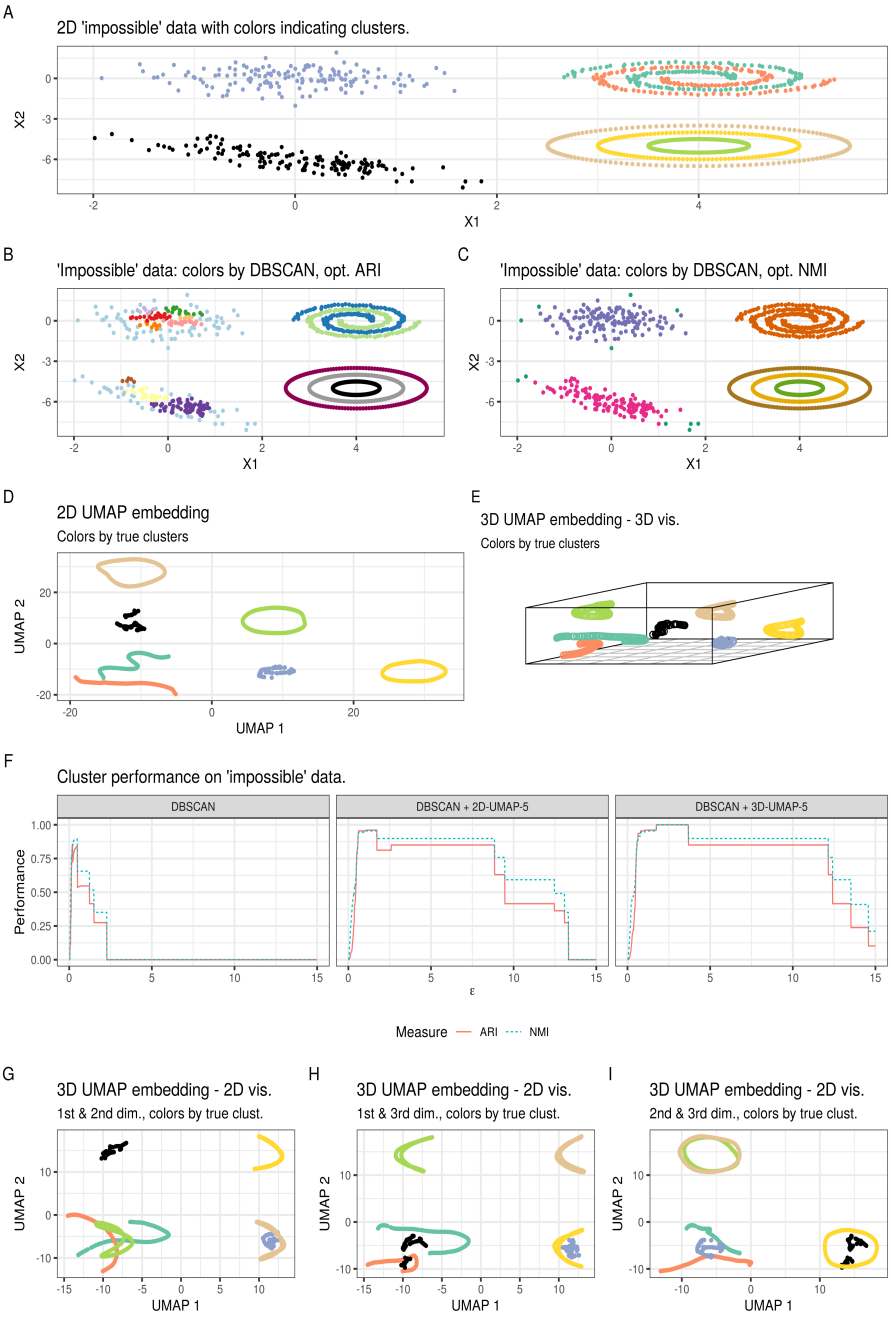

Fig. 3 Another example of complex synthetic data and the beneficial effect of "dimension inflation". 1st row: the "impossible" data with color according to true cluster structure. 2nd row: data colored according to DBSCAN cluster results if applied directly to the data (different optimal ε values for ARI and NMI). 3rd row: Visualizations of a 2D and 3D UMAP-5 embedding with colors according to true cluster structure. 4th row: ε-curves for DBSCAN applied to the data, a 2D UMAP-5, and a 3D UMAP-5 embedding. Last row: 2D visualizations of the 3D UMAP-5 embedding with colors according to true cluster structure. In all settings: DBSCAN computed for $\varepsilon \in [0.01, 15]$, step size: 0.01; $minPts = 5$.

However, the *three* dimensional UMAP-5 embedding not only further reduces parameter sensitivity but also allows for perfect cluster performances. A 3D visualization of this embedding is depicted in Figure 3 E, but note that a static 3D visualization does not make the improved separability visible very well. Figures 3 G-I show all pairwise plots of the three embedding dimensions of the 3D UMAP embedding, even though none of these 2D projections reflects the cluster structure well. We recommend basing exploratory analysis on 3D embeddings as they are more likely to yield good results in complex data than 2D embeddings and still allow for very reasonable visualizations with dynamic plotting tools.

4.2 Outliers and noise points

Outliers are another important property of a dataset, but their distinctiveness and relative isolation is unlikely to be preserved in their UMAP embeddings. Consider Figure 4 A & C, which shows two 2D datasets with two clusters and, firstly, with two outliers (in blue) on the left-hand side, and, secondly, with additional, uniformly distributed noise points (in grey) on the right-hand side. Corresponding UMAP embeddings for $k = 15$ are depicted in Figure 4 B & D. Although the cluster structure is preserved, in both cases the outliers are no longer detectable as such (note that no dimension reduction has taken place). Similarly for noise points, which are embedded into proximal clusters and then no longer detectable as noise.

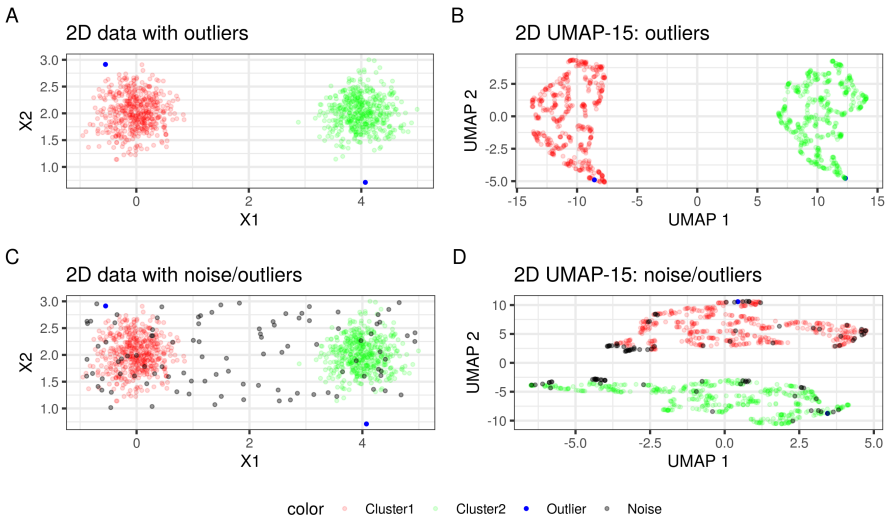

Fig. 4 Effect of UMAP on data with outliers and noise points. First column: 2D datasets with two clusters and two outliers (**A**) and two outliers and noise points (**C**). Second column: UMAP embeddings with $k = 15$ (**B** & **D**, respectively). The cluster structure is preserved. Outliers and noise points are forced into the clusters.

It has recently been shown for functional data that outlyingness can be seen as a metric structure of a dataset (Herrmann and Scheipl, 2021). Since UMAP does not preserve metric structure (i.e. distances) but connected components, the loss of the outlier structure is not surprising. Moreover, note that UMAP's local connectivity constraint, which ensures that each point is at least connected to its nearest neighbor, may render it generally impossible to preserve outlier structure in UMAP embeddings. Applying outlier detection methods in an additional preprocessing step before computing UMAP embeddings may solve this issue.

4.3 Overlapping and diffuse clusters

Clusters with considerable overlap or diffuse boundaries that result in a large likelihood of "bridge" points between nominally distinct clusters are especially challenging for most clustering algorithms.

First of all, consider Figure 5 A, which shows a 2D dataset consisting of two clusters that are connected by a small "bridge" of points (blue). From a purely topological perspective, we have a single connected topological subspace. A 2D UMAP representation, however, breaks the connected components apart, see Figure 5 B & C. Note, that this holds for a small value of $k = 15$ as well as for a very large value of $k = 505$. Another issue concerns clusters with substantial overlap, which are often modeled as diffuse components of a Gaussian mixture (Rasmussen, 2000). In such cases, UMAP and similar manifold learning methods are unlikely to improve clustering performance. Consider Figure 5 D. It shows a 2D dataset with two clusters following 2-dimensional Gaussian distributions with mean vectors $(0, 2)'$ and $(2, 2)'$ and unit covariance matrix. Note that in both embeddings (Figure 5 E & F) the clusters are not clearly separable, and the less so the larger UMAP's locality parameter k is chosen.

For strongly overlapping clusters, it is questionable to even consider such settings as ("pure") clustering tasks. From a topological perspective, such settings cannot be considered a well-posed clustering problem as there are no separable components in the data. However, in the presence of bridges, it seems reasonable to consider the dataset as consisting of two clusters. Whether overlapping clusters should be merged or considered separate must surely be answered w.r.t. the specific domain. Practitioners should be aware of how UMAP tends to behave in such settings: it typically breaks "bridges" apart and merges highly overlapping clusters.

4.4 Quantitative analysis of further synthetic data

In addition to the qualitative analyses of these toy datasets we investigate further examples quantitatively in this paragraph. The datasets under consideration are those from the Fundamental Clustering Problem Suite (FCPS) (Ultsch, 2005). These datasets are constructed such that they reflect specific clustering problems. Table 2 shows key characteristics of these datasets and the

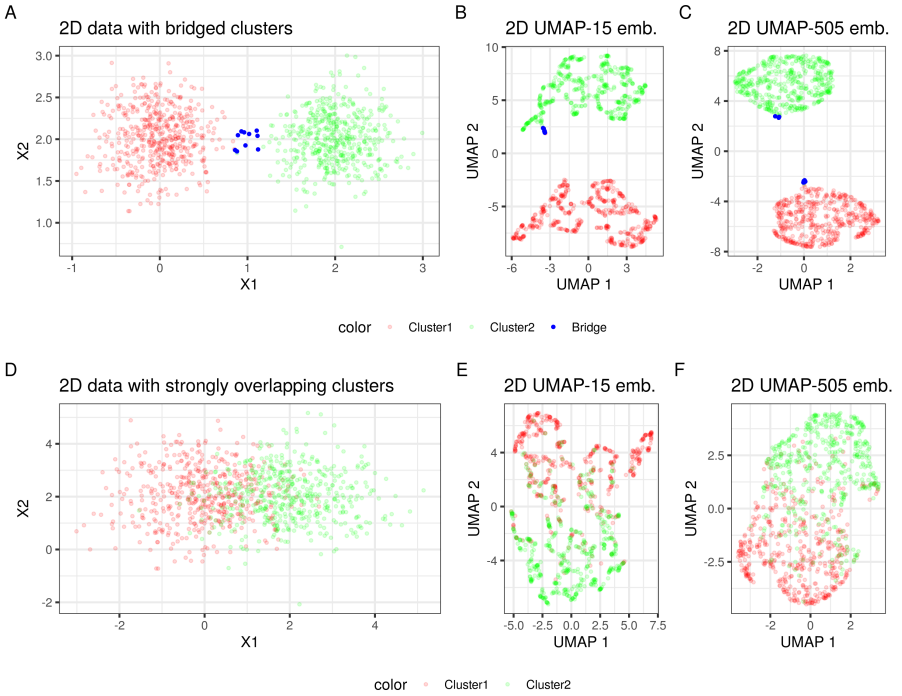

Fig. 5 Effect of UMAP on data with connected components. Upper row: 2D data with two bridged clusters. Lower row: 2D dataset with two strongly overlapping clusters. **A** & **D**: data. **B, C, E, F**: UMAP embeddings with $k = 15$ and $k = 505$, respectively. UMAP breaks the bridged components up into two clusters but does not break up the strongly overlapping components.

problems they present. More details including visualizations can be found in the corresponding papers (Thrun and Ultsch, 2020; Ultsch and Lötsch, 2020).

The results of applying DBSCAN directly to the data and on 2D UMAP embeddings with $k = 10$ are shown in Table 3. Depicted are the highest achievable ARI and NMI values by approach and dataset as well as the ε-range $\varepsilon_{[ARI>0]}$ for which ARI is greater than zero.

The results show that DBSCAN alone already yields perfect clustering performance for the datasets Hepta, Lsun, Chainlink, Atom, Target, WingNut, and GolfBall. However, note that UMAP clearly reduces ε sensitivity (much wider ε-range), i.e. it increases clusterability for Hepta, Lsun, Chainlink, Atom, Target.

On the datasets Tetra and TwoDiamonds, DBSCAN does not perform perfectly. These datasets represent problems (specified as "almost touching clusters" (Tetra) and "cluster borders defined by density" (TwoDiamonds)) with less clearly separable clusters. Consistent with the examples presented in section 3.1, inferring the topological structure via UMAP not only drastically

Table 2 Characteristics of the FCPS datasets: the number of clusters n_c, the number of observations n_{obs}, the number of features (dimensionality) p, and the problem as specified in corresponding papers (Thrun and Ultsch, 2020; Ultsch and Lötsch, 2020).

Name	n_c	n_{obs}	p	Problem
Hepta	7	212	3	different variances
LSun	3	400	3	different variances & inter cluster distances
Tetra	4	400	3	almost touching clusters
Chainlink	2	1000	3	not linearly separable
Atom	2	800	3	different variances & not linearly separable
EngyTime	2	4096	2	Gaussian mixture
Target	6	770	2	outliers
TwoDiamonds	2	800	2	cluster borders defined by density
Wingnut	2	1070	2	density vs. distance
Golfball	1	4002	3	no clusters at all

reduces ε sensitivity of DBSCAN, but it also improves clustering performance to (almost) perfect results in these examples.

In contrast to that, inferring the relevant structure is not possible with UMAP in the settings EngyTime and Target and thus it does not improve the performance of DBSCAN, it even reduces it. This is consistent with the results of the previous subsections: EngyTime is a setting with clusters that overlap strongly, while the Target data is a setting with six clusters of which four are defined by a few outliers.

Table 3 Maximum ARI and NMI and ε ranges corresponding to ARI > 0 for FCPS data.

	DBSCAN			UMAP + DBSCAN		
Data	ARI	NMI	$\varepsilon_{[ARI>0]}$	ARI	NMI	$\varepsilon_{[ARI>0]}$
Hepta	1	1	[0.0, 2.3]	1	1	[0.1, 19]
Lsun	1	1	[0.1, 0.7]	1	1	[0.1, 14]
Tetra	0.91	0.85	[0.2, 0.5]	0.99	0.99	[0.1, 7]
Chainlink	1	1	[0.0, 0.8]	1	1	[0.0, 7]
Atom	1	1	[0.8, 20]	1	1	[0.0, 13]
EngyTime	0.36	0.23	[0.0, 1]	0.29	0.26	[0.0, 0.9]
Target	1	0.97	[0.0, 2.3]	0.97	0.88	[0.0, 11]
TwoDiamonds	0.95	0.85	[0.0, 0.1]	1	1	[0.0, 4.7]
WingNut	1	1	[0.1, 0.3]	1	1	[0.0, 8.1]
GolfBall	1	NaN	[0.0, 20]	1	NA	[0.0, 20]

In summary, the synthetic examples investigated in this and the previous section show that inferring the topological structure of a dataset can dramatically improve and simplify clustering: improvement in the sense that cluster detection with DBSCAN is considerably more reliable, and simplification in the sense that finding good parameters for DBSCAN becomes significantly less challenging: the suitable ε-ranges are typically much wider, they consistently start near zero and ARI/NMI quickly reach their optimum in this range, so

that a quick and simple coarse grid search over small values of ε is likely to be successful.

We emphasize that these conclusions apply to diverse and challenging synthetic data settings that include low-dimensional as well as high-dimensional data, data with equal and unequal cluster densities, data with (many) irrelevant features, clusters of arbitrary shape, and not linearly separable clusters. In the next section, we show that this also holds for several real datasets.

5 Experiments on Real-World Data

An overview of the real datasets used in this study is given in Table 4. Since some of these datasets have already been used in other studies, we can investigate not only how the clustering performance of DBSCAN is improved if the topological structure of a dataset is inferred beforehand. We can additionally compare our results to those reported for other clustering methods. The set of datasets includes the well known Iris data (Anderson, 1935; Fisher, 1936), the Wine data (Aeberhard et al, 1994; Forina et al, 1988; Dua and Graff, 2017), the Pendigits data (Alimoğlu and Alpaydin, 2001; Dua and Graff, 2017) as well as the COIL (Nane et al, 1996), MNIST (Lecun et al, 1998) and fashion MNIST (FMNIST) (Xiao et al, 2017) data. Following Mukherjee et al (2019), we use two different versions of FMNIST: one with the original ten clusters and a version reduced to five clusters which are pooled from the original ten based on their similarity. The results of applying DBSCAN directly to the datasets and to the embeddings obtained with UMAP are depicted in Figure 6 and Table 5.

Table 4 Characteristics of the real datasets: the number of clusters n_c, the number of observations n_{obs}, and the number of features (dimensionality) p. As in the ClusterGAN paper (Mukherjee et al, 2019) we investigate two versions of FMNIST: FMNIST-10 and FMNIST-5, the clusters in the latter are: 1: Tshirt/Top, Dress; 2: Trouser; 3: Pullover, Coat, Shirt; 4: Bag; 5: Sandal, Sneaker, Ankle Boot.

Name	n_c	n_{obs}	p
Iris	3	150	5
Wine	3	176	14
COIL	20	1440	16385
Pendigits	10	10992	17
MNIST	10	70000	784
FMNIST-10	10	70000	784
FMNIST-5	5	70000	784

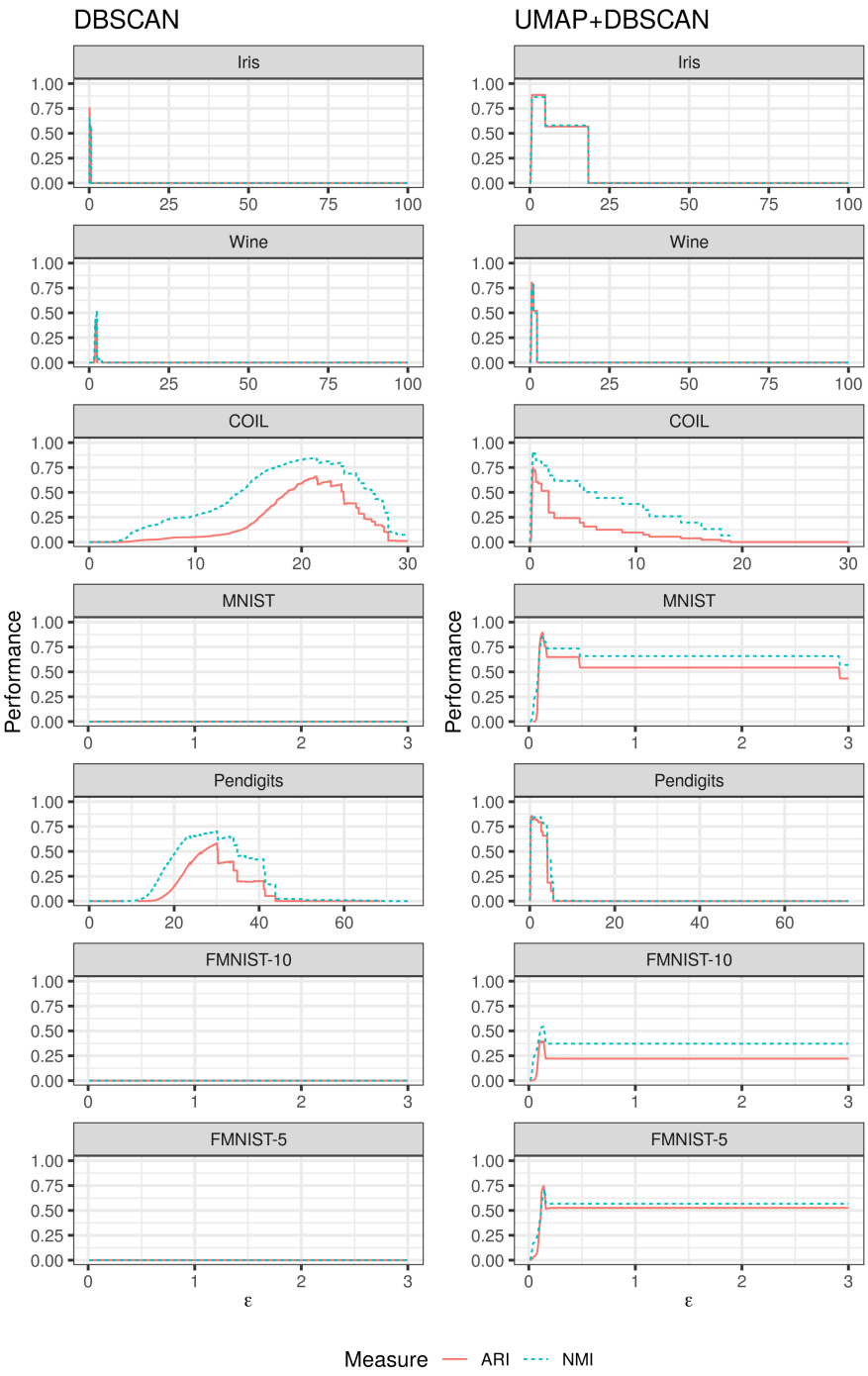

Fig. 6 ARI and NMI as functions of ε for the real datasets. Parameters: $k = 10$ and $d = 3$ (UMAP); $minPts = 5$, ε-step-size $= 0.01$ (DBSCAN).

Figure 6 shows ARI and NMI as a function of ε for the different datasets. Table 5 details the optimum ARI and NMI achieved within the considered ε-ranges. We inferred the topological structure of the datasets for three different values of $k \in \{5, 10, 15\}$. Note that we did not tune UMAP at all and used `min_dist` $= 0.1$, `n_components` $= 3$ and spectral initialization throughout. Iris and Wine data features were scaled respectively standardized.

In general, the results show that what has been observed for the synthetic examples also holds for real data. For all considered settings, inferring the topological structure of the dataset via UMAP before applying DBSCAN leads to better clustering performances than applying DBSCAN directly, dramatically so for MNIST and FMNIST. Moreover, it reduces ε sensitivity of DBSCAN with suitable ε-ranges starting close to zero and with high (> 0.5) ARI and NMI values for large parts of the ε-range. For DBSCAN directly applied to (F)MNIST, we additionally scanned the ε-range $[0, 100]$ with a step size of 0.1, but performance did not improve over this extended search grid.

We also investigate the effect of optimizing the separability by constructing embedding vectors instead of using the fuzzy edge weights directly for datasets Iris, Wine, COIL, and Pendigits. Clustering using UMAP's fuzzy graph weights directly performs worse, as expected. For example on the Iris data, computing embedding vectors with UMAP-10 leads to optimal ARI/NMI $= 0.89/0.86$ over an ε-range of $[0.67, 4.82]$ in contrast to $0.88/0.84$ over $[0.6, 0.61]$ if only the fuzzy graph weights of UMAP-10 are used. Both variants still yield better results than applying DBSCAN directly to the data (optimal ARI/NMI $= 0.75/0.67$). We found similar results for Wine, COIL, and Pendigits, see appendix B.

Table 5 Maximum ARI and NMI for the real datasets. DBSCAN directly applied to the data and to 3D UMAP embeddings for $k \in \{5, 10, 15\}$. For the explored ε-ranges, see Fig. 6.

	DBSCAN		DBS+UMAP-5		DBS+UMAP-10		DBS+UMAP-15	
	ARI	NMI	ARI	NMI	ARI	NMI	ARI	NMI
Iris	0.75	0.67	0.70	0.75	0.89	0.86	0.89	0.86
Wine	0.44	0.52	0.81	0.77	0.81	0.78	0.80	0.79
Pendigits	0.58	0.70	0.80	0.82	0.86	0.85	0.83	0.85
COIL	0.66	0.85	0.82	0.93	0.75	0.91	0.70	0.88
MNIST	0.00	0.00	0.69	0.70	0.90	0.85	0.87	0.85
FMNIST-10	0.00	0.00	0.41	0.59	0.40	0.54	0.38	0.54
FMNIST-5	0.00	0.00	0.60	0.62	0.75	0.71	0.63	0.63

In addition, our results show that the fast, simple and very easily tuneable approach we have proposed leads to comparable or superior clustering performances than recently proposed clustering methods such as ClusterGAN (Mukherjee et al, 2019) and SPECTACL(N) (Hess et al, 2019) in some settings. Table 6 lists the highest results obtained on the respective datasets in other studies (Goebl et al, 2014; Mautz et al, 2017; Mukherjee et al, 2019; Hess et al, 2019). On Pendigits and FMNIST-5, DBSCAN applied to UMAP embeddings

performs better than the best-performing methods FOSSCLU and ClusterGAN as reported by Goebl et al (2014), Mautz et al (2017), and Mukherjee et al (2019). On MNIST, comparable performance is achieved w.r.t. ClusterGAN and better performance w.r.t. SPECTACL(N). Only for the Wine data and FMNIST-10 are better performance reported for methods FOSSCLU, LDA-k-means, and ClusterGAN.

Table 6 Optimal ARI and NMI for some of the real datasets reported in other studies and the methods used. The last two columns show the corresponding optimal performances achieved with DBSCAN & UMAP.

Study	Conf.	Data	ARI	NMI	Method(s)	ARI (DBS+UMAP)	NMI (DBS+UMAP)
Goebl et al, 2014	IEEE	Pendigits	NA	0.77	FOSSCLU	0.86	0.85
		Wine	NA	0.87	FOSSCLU	0.80	0.79
Mautz et al, 2017	KDD	Pendigits	NA	0.77	FOSSCLU	0.86	0.85
		Wine	NA	0.93	LDA-k-means	0.80	0.79
Mukherjee et al, 2019	AAAI	Pendigits	0.65	0.73	ClusterGAN	0.86	0.85
		MNIST	0.89	0.90	ClusterGAN	0.90	0.85
		FMNIST-10	0.50	0.64	ClusterGAN	0.41	0.59
		FMNIST-5	0.48	0.59	ClusterGAN, GAN with bp	0.75	0.71
Hess et al, 2019	AAAI	MNIST	NA	0.76	SPECTACL(N)	0.90	0.85

It must be emphasized that these methods also require analysts to pre-specify a fixed number of clusters that are to be found. ClusterGAN's optimal performances reported in Table 6 were achieved only if the true number of clusters was supplied (Mukherjee et al, 2019). The performance on MNIST considerably deteriorated if the number of clusters was not correctly specified. Recall that one of the major advantages of DBSCAN is that it does not require pre-specifying the number of clusters, in contrast to the complexity of specifying and training ClusterGAN. It should be taken into account, first of all, that a suitable network architecture needs to be defined. Note that standard architectures specified elsewhere had to be adapted for ClusterGAN to achieve satisfactory performance. In addition, the various hyperparameters for the GAN, the SGD optimizer, and the generator-discriminator updating require substantial tuning. Finally, note that our approach works well in settings with both few and many clusters and for both small and large numbers of observations. This is also in contrast to ClusterGAN, which was "particularly difficult [... to train ...] with only a few thousand data points" (Mukherjee et al, 2019, p. 4616).

6 Discussion

In summary, the presented results show that considering clustering from a topological perspective consistently simplified analysis and improved results in a wide range of settings: from a practical perspective, inferring the topological structure of datasets and representing this structure in suitable embedding

vectors that are, in some sense, optimized for separability between the different connected components (dramatically) increased clustering performances of DBSCAN, even outperforming a highly complex deep learning-based clustering method, as long as the clusters did not exhibit large overlap. These insights suggest some conceptual conclusions and raise a number of fundamental questions for cluster analysis, which we will discuss in the following.

To begin with, we argue that two "perspectives" on cluster analysis should be more strictly distinguished: on the one hand, settings where the aim is to infer the number of connected components in a dataset (the "topological perspective"), and on the other hand, settings where clusters may show considerable overlap (in the following the "probabilistic perspective"). If the "perspective" (implicitly) taken is not clearly specified, the results of cluster analysis can be misleading. For example, in applied, exploratory analyses relevant information may be lost, while in methodological analyses method comparisons can be misleading.

Consider a truly unsupervised and exploratory setting (i.e. the true number of clusters is not known and determining it is a crucial part of the problem) in an applied context. From the "topological perspective" applying methods that yield a fixed, pre-specified number of clusters is highly questionable in this situation. If the number of clusters is determined a priori for example via domain knowledge, the analysis cannot falsify these a priori assumptions about the data and may hide any unexpected structure. This seems contradictory to the purpose of an exploratory analysis, where the discovery of unexpected structures can yield valuable new insights. If, on the other hand, approaches such as elbow-plots of cluster quality metrics are used to determine the number of clusters n_c in a data-driven way, methods inferring and enhancing connected components should be used in the first place.

Another issue concerns the evaluation of competing methods for clustering using datasets with label information. Label information can be misleading, in particular, if it is (also) used to pre-specify n_c, as the label information may not be consistent with the unconnected components of a dataset. Consider the FMNIST example, where a simple modification of label information – merging the original 10 into 5 broader categories – leads to considerably different results. Note that this change of labels was not introduced here, but in Mukherjee et al (2019). We assume that the performance of ClusterGAN on FMNIST – as measured based on the original labels – was not as convincing as for the other datasets. Since it requires no specialized domain knowledge to assess the general similarity of clusters in this dataset containing images of pieces of apparel, a change of labels is easy to do. But while this change did not improve the performance of ClusterGAN in terms of ARI and NMI by much, it considerably improves the performance of DBSCAN + UMAP. In other words: the labels were presumably changed such that they were much more consistent with the actual unconnected components – i.e. clusters – in the data. If only the original ten categories of clothing had been considered here, the method comparison would have been misleading, as the different ability

of the methods to identify the (un)connected components of the data would have gone unnoticed. The original label information arguably does not reflect the actual cluster structure of the data. This is likely to be the case in many labeled datasets.

On the other hand, consider settings with overlapping clusters. Taking the topological perspective does not make a lot of sense here, as there are no unconnected components if clusters (strongly) overlap, and our investigations showed that it is, in general, questionable that it is possible to infer such cluster structure with methods that aim to infer connected components. In such settings, one should rather take a "probabilistic perspective" and assume that the data follow a joint multi-modal probability distribution, i.e., a mixture of probability distributions. Note that this usually implies some kind of domain knowledge from which it makes sense to assume such structure. Many prominent clustering methods such as k-means, Gaussian Mixture models, or approaches based on the EM algorithm are based on this perspective. It has to be emphasized that our experiments on several widely used real-world benchmark datasets showed that an approach based on the topological perspective, which does not use the true number of clusters as a parameter, can perform comparable or even better than methods that do so.

These considerations raise some important questions. First of all, from a rather practical perspective: Is it fair to compare methods that require n_c as a parameter with those that do not? How trustworthy is the widely used approach to evaluate clustering methods using labeled data? Is it at all useful to apply non-probabilistic clustering methods on data with assumed strong cluster overlap?

Moreover, from a rather general conceptual perspective: Can there be methods that work optimally both in settings with large cluster overlap and settings of high separability? As Schubert et al (2017, p. 19) state in that regard:

> "To get deeper insights into DBSCAN, it would also be necessary to evaluate with respect to utility of the resulting clusters, as our experiments suggest that the datasets used do not yield meaningful clusters. We may thus be benchmarking on the 'wrong' datasets (but, of course, an algorithm should perform well on any data)."

This already points to the problem of "wrong" datasets, while on the other hand, they state a method should perform well in any setting. In the light of the insights presented here, we would argue that it is very fruitful to investigate the characteristics of settings in which a method or combination of methods works specifically well or even optimally. As outlined, we consider in particular high cluster overlap in contrast to well separable clusters examples of such settings. The underlying principles are fundamentally different (disconnected domains of the clusters vs. connected domains of the clusters) and may require different, maybe even contradictory objectives to be optimized. This is specifically relevant as a dataset may consist of both sorts of (assumed) structures. We think the insights and results presented here support this view.

7 Conclusion

This work considered cluster analysis from a topological perspective. Our results suggest that the crucial issue in clustering is not the nominal dimension of the dataset or whether it contains many irrelevant features, but rather how separable the clusters are in the ambient observation space they are embedded in. Extensive experiments on synthetic and real datasets clearly show that focusing on the topological structure of the data can dramatically improve and simplify cluster analysis both in low- and high-dimensional settings. To demonstrate this principle in practice, we used the manifold learning method UMAP to infer the connected components of the datasets and to create embedding vectors optimized for separability, to which we then applied DBSCAN.

Using synthetic data, we showed that this makes results much more robust to hyperparameters in a diverse set of problems including low-dimensional as well as high-dimensional data, data with equal and unequal cluster densities, data with (many) irrelevant features and clusters of arbitrary, not linearly separable shapes. The parameter sensitivity of DBSCAN is consistently and dramatically reduced, simplifying the search for a suitable ε-value. Moreover, the cluster detection performance of DBSCAN was considerably improved compared to applying it directly to the data.

Experiments in real data settings corroborated these insights. In addition, our results showed that the simple approach of combining UMAP and DBSCAN can even outperform complex clustering methods SPECTACL and deep-learning-based ClusterGAN on complex image data such as Fashion MNIST.

All these results were obtained with very little hyperparameter tuning for UMAP. In particular, we always used a small value of the parameter k/`n_neighbors` – $k \in \{5, 10, 15\}$ in most of our experiments – markedly reducing the complexity of the parameter choice in density-based clustering. All other parameters were set to the default values. Based on a simple toy example we provided a detailed technical explanation of why the choice of a small k is reasonable for the purpose of clustering.

Finally, we propose a conceptual differentiation of cluster analysis suggested by the topological perspective and the presented results. Specifically, we argue that settings with high cluster overlap in contrast to well separable clusters should be considered as fundamentally different settings which require different kinds of methods for optimal results, a distinction usually not made explicit enough. We also propose that using external label information to evaluate clustering solutions should only be done if these labels actually correspond to the (un)connected components of the data manifold from which observations are sampled. If this is not the case, we would argue that evaluation metrics diverge from what clustering algorithms should properly optimize for – identifying (un)connected components – and results will be misleading.

We think these considerations point out important questions to be investigated in future work.

Declaration

Funding

This work has been funded by the German Federal Ministry of Education and Research (BMBF) under Grant No. 01IS18036A. The authors of this work take full responsibility for its content.

Conflict of Interest

The authors have no competing interests to declare that are relevant to the content of this article.

Data and code availability

All code and data to reproduce the results can be found on GitHub: https://github.com/HerrMo/topoclust.

Appendix A Embedding variability

In section 3.2, we showed that although the computation of embedding vectors induces some variability with respect to the meaningful ε-range, it also leads to considerably improved separability and is therefore crucial from a clustering perspective. Here we provide additional experiments on this which are based on the synthetic settings from section 3.1 and the three smallest ($n_{\text{obs}} < 10^4$ observations) real datasets Iris, Wine, and COIL. We computed 25 embeddings for each of the datasets (and k values in the case of the real datasets) and corresponding clusterings on ε-grids $[0.01, 15]$ and $[0.01, 25]$, respectively, with a step size of 0.01. For each ε-value, the individual minimal, mean, and maximal ARI and NMI values are computed over the 25 replications. Figures A1 and A2 depict the corresponding minimum, mean, and maximum ARI and NMI curves. Note that the curves do not reflect a single embedding, but the worst/mean/best case over all 25 embeddings for each individual ε-value. In addition, the maximum ARI and NMI values obtained by applying DBSCAN directly to the data are shown as a black dashed horizontal line and the corresponding ε-value as a black dashed vertical line.

In summary, the results again show that optimizing embedding vectors induces some variability with respect to the sensible ε-range across different embeddings. However, this variability can be neglected if the main focus is on improving cluster detection. First of all, the variability does not affect the fact that the sensible ε-ranges start near zero and quickly reach the optimal value, which is in stark contrast to DBSCAN directly applied to the data (see the black dashed horizontal and vertical lines, and Fig. 1). In addition, in all settings, the mean ARI and NMI curves are higher on larger parts of the ε-ranges as the maximum ARI and NMI for DBSCAN directly applied to the data. Note that, except for UMAP-5 on Iris and UMAP-15 on COIL, this holds for the minimum curves as well!

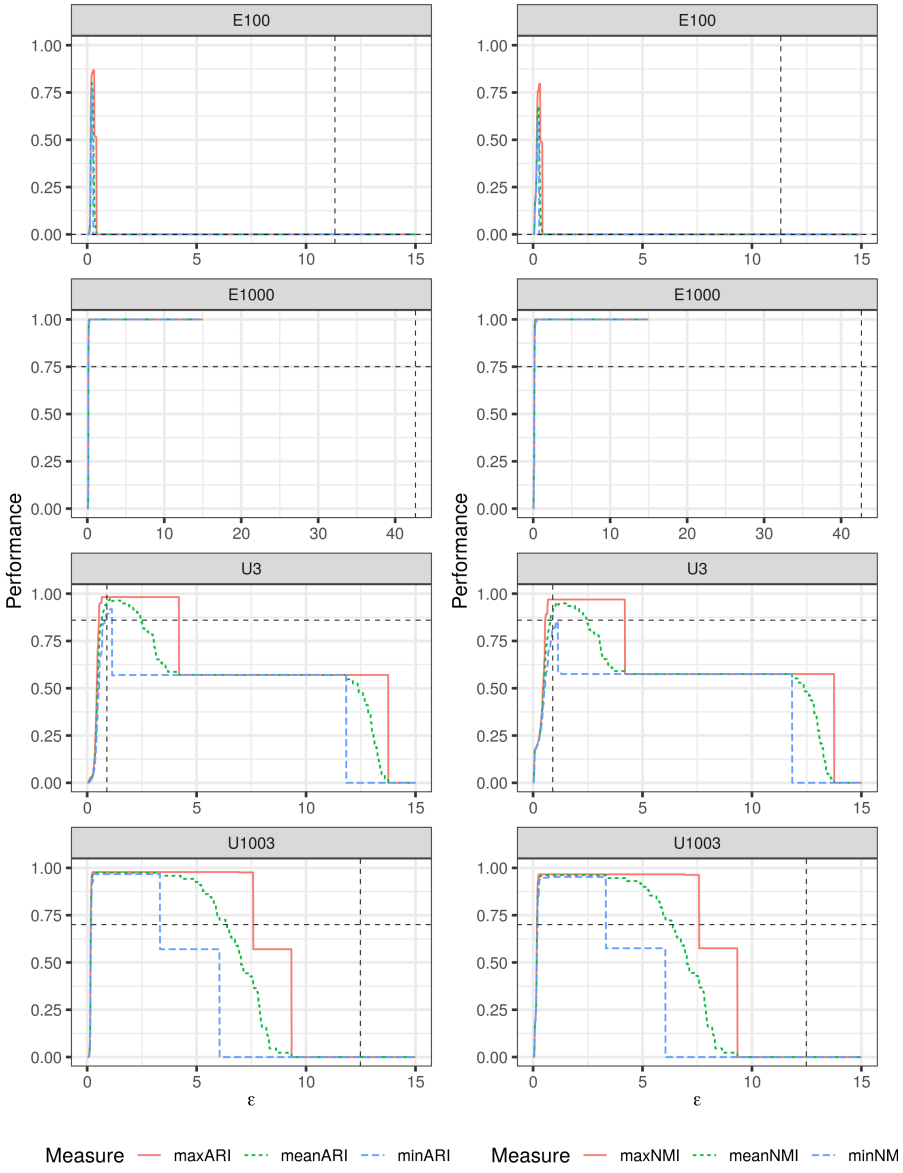

Fig. A1 Maximum, mean, and minimum ARI (left column) and NMI (right column) curves summarized over 25 embeddings of the four synthetic settings $E_{100}, E_{1000}, U_3, U_{1003}$. Note, the curves do not reflect a single embedding, but the worst/mean/optimal case over all 25 embeddings for each individual ε-value. The maximum ARI and NMI values obtained by applying DBSCAN directly to the data are shown as a black dashed horizontal line and the corresponding ε-value as a black dashed vertical line. DBSCAN computed for $\varepsilon \in [0.01, 15]$, step size: 0.01; $minPts = 5$.

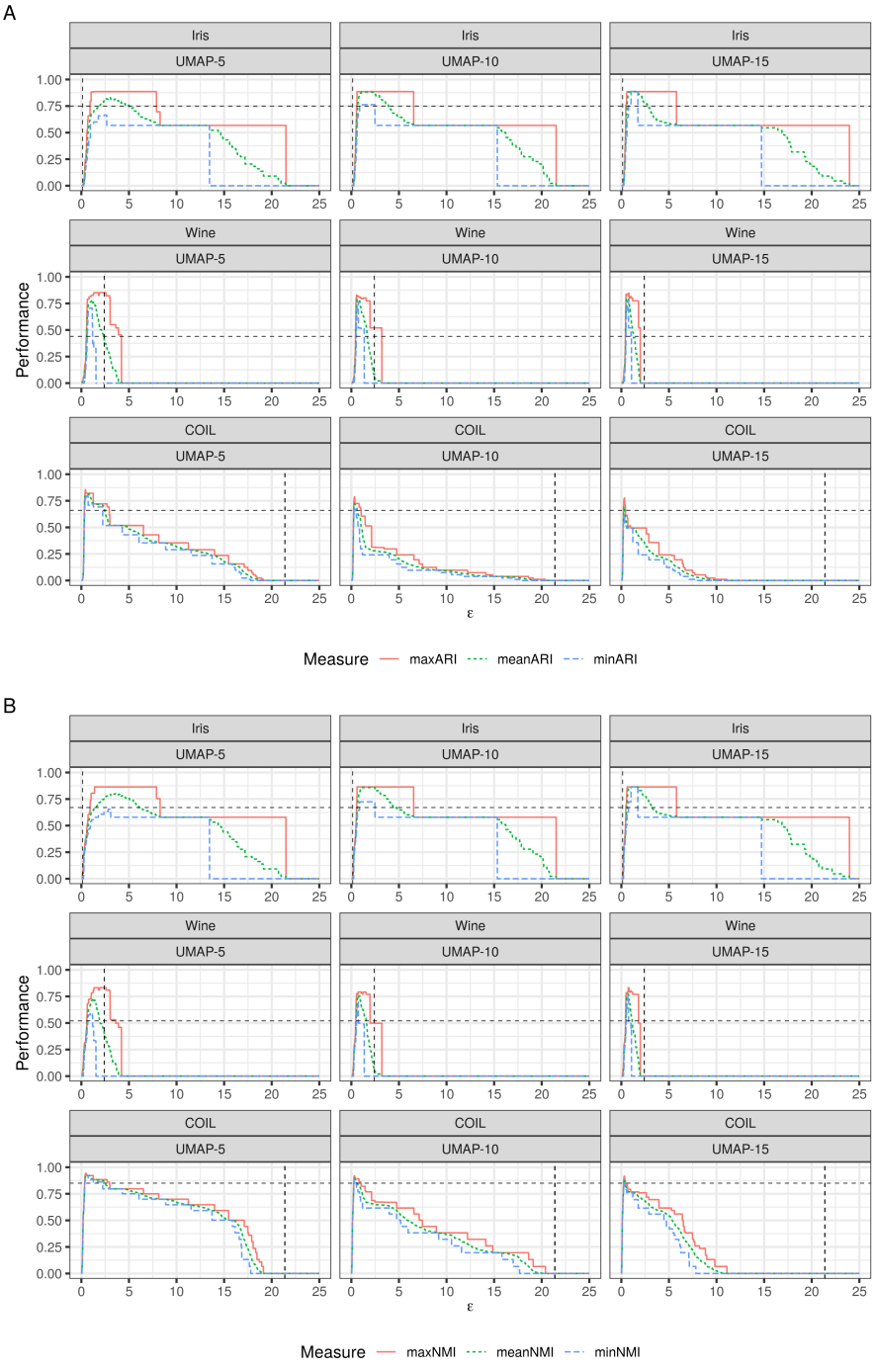

Fig. A2 Maximum, mean, and minimum ARI (A) and NMI (B) curves summarized over 25 embeddings of the Iris, Wine, and COIL data. Note, the curves do not reflect a single embedding, but the worst/mean/optimal case over all 25 embeddings for each individual ε-value. The maximum ARI and NMI values obtained by applying DBSCAN directly to the data are shown as a black dashed horizontal line and the corresponding ε-value as a black dashed vertical line. DBSCAN computed for $\varepsilon \in [0.01, 25]$, step size: 0.01; $minPts = 5$.

Appendix B Using just the fuzzy graph weights versus using embedding vectors

Figure B3 shows ARI and NMI as a function of ε for four of the real datasets. Cluster results were computed using just the fuzzy graph weights, without additionally computing embedding vectors. Converting the graph weights into dissimilarities via $d_{ij} = 1 - v_{ij}$, $i \neq j$, means that the meaningful ε-range is restricted to $[0, 1]$. Moreover, the sensible ε-ranges (yielding optimal or high ARI/NMI values) are smaller than those resulting based on additionally optimized embedding vectors.

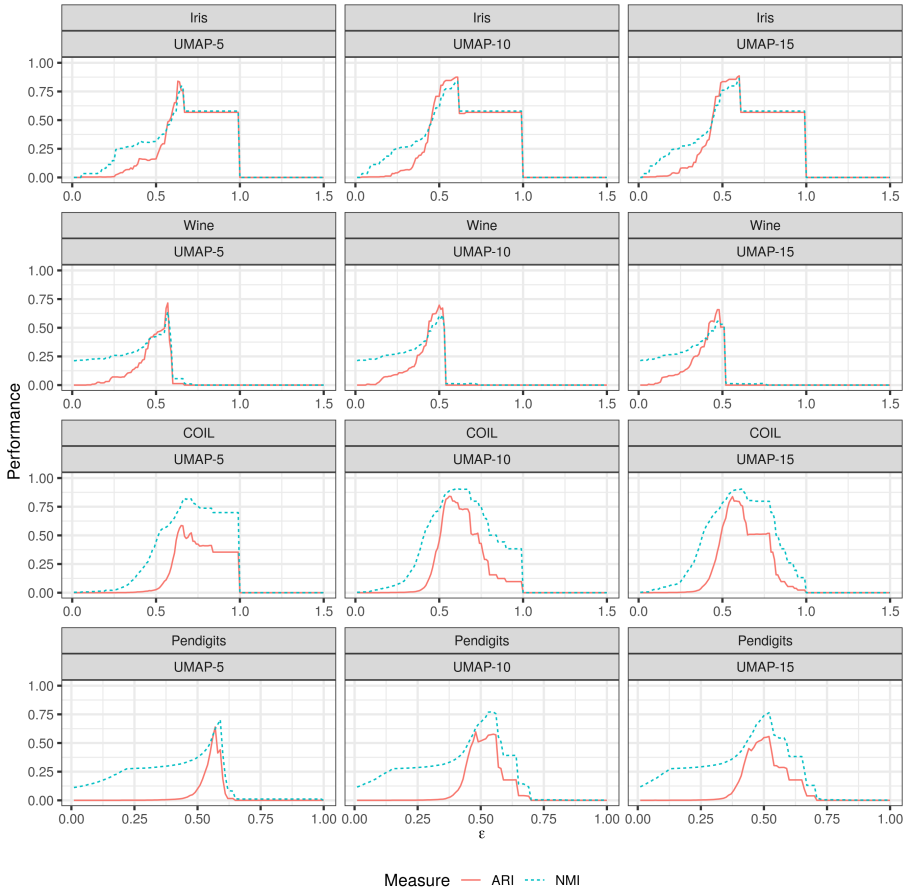

Fig. B3 ARI and NMI as a function of ε for four of the real datasets. Results obtained by applying DBSCAN on the fuzzy graph computed by UMAP (converted into a dissimilarity matrix via $d_{ij} = 1 - v_{ij}$, $i \neq j$, with v_{ij} an edge weight). Embedding vectors optimized for separability have not been constructed. DBSCAN computed for $\varepsilon \in [0.01, 1.5]$, step size: 0.01; $minPts = 5$.

Here we shortly detail this effect for the Wine, COIL, and Pendigits data based on the UMAP-10 results. The Iris data results are exemplarily discussed in section 5.

For the Wine data, only computing the fuzzy graph with UMAP-10 leads to optimal ARI/NMI = 0.7/0.61 for a single $\varepsilon = 0.5/0.52$. In contrast, additionally computing optimized embedding vectors leads to ARI/NMI = 0.81/0.78 for $\varepsilon \in [0.64, 0.69]/[1.11, 1.16]$. Unlike the Iris and Wine data, the optimal ARI/NMI value for the Pendigits and COIL data is only achievable for a single ε-value. Using embedding vectors is nevertheless beneficial. To see this, consider that on Pendigits an ARI/NMI > 0.6 can be obtained over $[0.17, 4.04]/[0.16, 4.13]$ with embedding vectors. Only using the fuzzy graph would mean that an ARI > 0.6 is not at all achievable and NMI > 0.6 only for $\varepsilon \in [0.48, 0.56]$. Similar holds for COIL, with ARI/NMI > 0.6 for $\varepsilon \in [0.25, 0.8]/[0.18, 4.07]$ in contrast to $[0.52, 0.68]/[0.45, 0.79]$.

Again, it needs to be emphasized that only using the fuzzy graph still yields better results than applying DBSCAN directly to the data. For example, applying DBSCAN directly to the Wine data yields optimal ARI/NMI = 0.44/0.52.

In summary, these investigations also show that computing embedding vectors optimized for separability on top of the fuzzy graph not only reduces parameter sensitivity of the clustering method but can also lead to a better clustering performance due to improved separability.

Appendix C Real data embedding visualizations

Figure C4 shows 2D UMAP-10 embeddings of the real datasets under investigation. Colors correspond to the class labels. As can be seen, the inferred connected components clearly agree with the labels for most of the datasets. In FMNIST this holds much better for the 5-label-set. However, note that although 3D embeddings are used in the experiments as they are better suited for cluster detection, they are less well suited for static visualizations (see section 4). That is why we depict UMAP-10 embeddings optimized in two dimensions (i.e. $d = 2$) here.

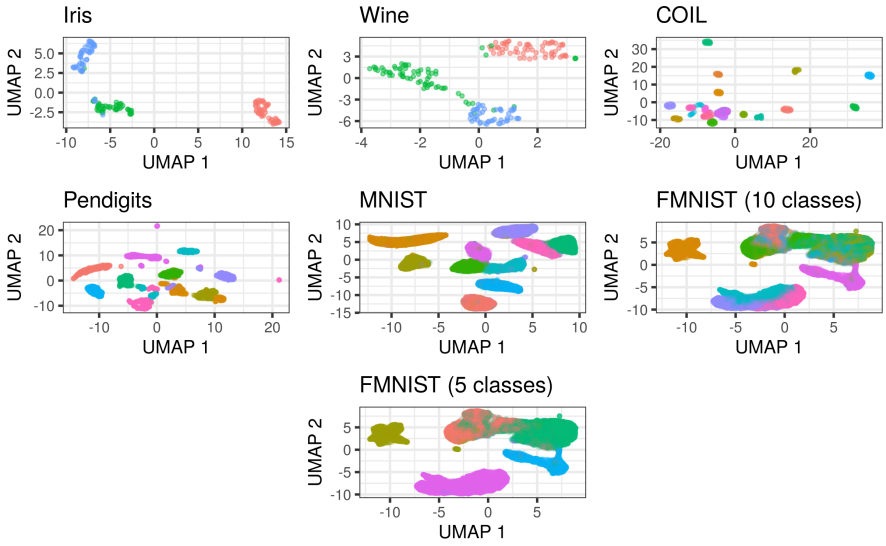

Fig. C4 Visualizing 2D UMAP-10 embeddings of the real datasets. Note that an embedding dimension of $d = 2$ was chosen for the purpose of optimal static visualization, in contrast to $d = 3$ used for better cluster detection in the quantitative experiments in section 5.

References

Aeberhard S, Coomans D, de Vel O (1994) Comparative analysis of statistical pattern recognition methods in high dimensional settings. Pattern Recognit 27(8):1065–1077. https://doi.org/10.1016/0031-3203(94)90145-7

Aggarwal CC (2014) An introduction to cluster analysis. In: Aggarwal CC, Reddy CK (eds) Data Clustering, 1st edn. Chapman and Hall/CRC, Boca Raton, p 1–28, https://doi.org/10.1201/9781315373515

Aggarwal CC (2015) Data Mining: The Textbook. Springer, Cham, https://doi.org/10.1007/978-3-319-14142-8

Aggarwal CC, Reddy CK (eds) (2014) Data clustering: Algorithms and Applications. Chapman and Hall/CRC, Boca Raton, https://doi.org/10.1201/9781315373515

Alimoğlu F, Alpaydin E (2001) Combining multiple representations for pen-based handwritten digit recognition. Turk J Elec Engin & Comp Sci 9(1)

Allaoui M, Kherfi ML, Cheriet A (2020) Considerably improving clustering algorithms using UMAP dimensionality reduction technique: A comparative study. In: International Conference on Image and Signal Processing, Springer, Cham, pp 317–325, https://doi.org/10.1007/978-3-030-51935-3_34

Anderson E (1935) The irises of the gasp peninsula. Bull Am Iris Soc 59:2–5

Ankerst M, Breunig MM, Kriegel HP, et al (1999) OPTICS: Ordering points to identify the clustering structure. ACM SIGMOD Rec 28(2):49–60. https://doi.org/10.1145/304181.304187

Arias-Castro E, Lerman G, Zhang T (2017) Spectral clustering based on local PCA. J Mach Learn Res 18(9):1–57. URL http://jmlr.org/papers/v18/14-318.html

Assent I (2012) Clustering high dimensional data. WIREs Data Min Knowl Discov 2(4):340–350. https://doi.org/10.1002/widm.1062

Barton T, Bruna T, Kordik P (2019) Chameleon 2: An improved graph-based clustering algorithm. ACM Trans Knowl Discov Data 13(1). https://doi.org/10.1145/3299876

Bengio Y, Courville A, Vincent P (2013) Representation learning: A review and new perspectives. IEEE Trans Pattern Anal Mach Intell 35(8):1798–1828. https://doi.org/10.1109/TPAMI.2013.50

Beyer K, Goldstein J, Ramakrishnan R, et al (1999) When is nearest neighbor meaningful? In: Beeri C, Buneman P (eds) Database Theory ICDT99. ICDT 1999. Lecture Notes in Computer Science, vol 1540. Springer, Berlin, Heidelberg, pp 217–235, https://doi.org/10.1007/3-540-49257-7_15

Boudiaf M, Rony J, Ziko IM, et al (2020) A unifying mutual information view of metric learning: Cross-entropy vs. pairwise losses. In: Vedaldi A, Bischof H, Brox T, et al (eds) Computer Vision ECCV 2020. ECCV 2020. Lecture Notes in Computer Science, vol 12351. Springer, Cham, pp 548–564, https://doi.org/10.1007/978-3-030-58539-6_33

Campello RJ, Moulavi D, Sander J (2013) Density-based clustering based on hierarchical density estimates. In: Pei J, Tseng V, Cao L, et al (eds) Advances in Knowledge Discovery and Data Mining. PAKDD 2013. Lecture Notes in Computer Science, vol 7819. Springer, Berlin, Heidelberg, pp 160–172, https://doi.org/10.1007/978-3-642-37456-2_14

Cayton L (2005) Algorithms for manifold learning. Tech. rep., University of California at San Diego

Chazal F, Michel B (2021) An introduction to topological data analysis: Fundamental and practical aspects for data scientists. Front Artif Intell 4. https://doi.org/10.3389/frai.2021.667963

Dalmia A, Sia S (2021) Clustering with UMAP: Why and how connectivity matters. arXiv preprint URL https://arxiv.org/abs/2108.05525

Dempster AP, Laird NM, Rubin DB (1977) Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc Series B Stat Methodol 39(1):1–22. URL http://www.jstor.org/stable/2984875.

Doraiswamy H, Tierny J, Silva PJ, et al (2021) TopoMap: A 0-dimensional homology preserving projection of high-dimensional data. IEEE Trans Vis Comput Graph 27(2):561–571. https://doi.org/10.1109/TVCG.2020.3030441

Dua D, Graff C (2017) UCI Machine Learning Repository. URL http://archive.ics.uci.edu/ml

Ester M, Kriegel HP, Sander J, et al (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, pp 226–231

Feldman D, Schmidt M, Sohler C (2020) Turning big data into tiny data: Constant-size coresets for k-means, PCA, and projective clustering. SIAM J Comput 49(3):601–657. https://doi.org/10.1137/18M1209854

Fisher RA (1936) The use of multiple measurements in taxonomic problems. Ann Eugen 7(2):179–188

Forina M, Leard R, Armanino C, et al (1988) Parvus: an extendible package for data exploration, classification and correlation. Institute of Pharmaceutical and Food Analysis and Technologies, Via Brigata Salerno, 16147 Genoa, Italy

Giordani P, Ferraro MB, Martella F (2020) An Introduction to Clustering with R, 1st edn. Springer, Singapore, https://doi.org/10.1007/978-981-13-0553-5

Goebl S, He X, Plant C, et al (2014) Finding the optimal subspace for clustering. In: 2014 IEEE International Conference on Data Mining, pp 130–139, https://doi.org/10.1109/ICDM.2014.34

Guan S, Loew M (2021) A novel intrinsic measure of data separability. arXiv preprint URL https://arxiv.org/abs/2109.05180

Hastie T, Tibshirani R, Friedman J (2009) The Elements of Statistical Learning: Data Mining, Inference and Prediction, 2nd edn. Springer, New York, https://doi.org/10.1007/978-0-387-84858-7

Hennig C, Meila M, Murtagh F, et al (2015) Handbook of Cluster Analysis, 1st edn. Chapman and Hall/CRC, New York, https://doi.org/10.1201/b19706

Herrmann M, Scheipl F (2021) A geometric perspective on functional outlier detection. Stats 4(4):971–1011. https://doi.org/10.3390/stats4040057

Hess S, Duivesteijn W, Honysz P, et al (2019) The SpectACl of nonconvex clustering: A spectral approach to density-based clustering. In: Proceedings of the AAAI conference on artificial intelligence, pp 3788–3795, https://doi.org/10.1609/aaai.v33i01.33013788

Hozumi Y, Wang R, Yin C, et al (2021) UMAP-assisted k-means clustering of large-scale SARS-CoV-2 mutation datasets. Comput Biol Med 131:104,264. https://doi.org/10.1016/j.compbiomed.2021.104264

Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1):193–218. https://doi.org/10.1007/BF01908075

Jain AK (2010) Data clustering: 50 years beyond k-means. Pattern Recognit Lett 31(8):651–666. https://doi.org/10.1016/j.patrec.2009.09.011

Jain AK, Murty MN, Flynn PJ (1999) Data clustering: a review. ACM Comput Surv 31(3):264–323. https://doi.org/10.1145/331499.331504

Kaya IE, Pehlivanlı AÇ, Sekizkardeş EG, et al (2017) PCA based clustering for brain tumor segmentation of T1w MRI images. Comput Methods Programs Biomed 140:19–28. https://doi.org/10.1016/j.cmpb.2016.11.011

Kobak D, Linderman GC (2021) Initialization is critical for preserving global data structure in both t-SNE and UMAP. Nat Biotechnol 39(2):156–157. https://doi.org/10.1038/s41587-020-00809-z

Kriegel HP, Kröger P, Zimek A (2009) Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Trans Knowl Discov Data 3(1). https://doi.org/10.1145/1497577.1497578

Kriegel HP, Kröger P, Sander J, et al (2011) Density-based clustering. WIREs Data Min Knowl Discov 1(3):231–240. https://doi.org/10.1002/widm.30

Lecun Y, Bottou L, Bengio Y, et al (1998) Gradient-based learning applied to document recognition. In: Proceedings of the IEEE, pp 2278–2324, https://doi.org/10.1109/5.726791

Lee JA, Verleysen M (2007) Nonlinear Dimensionality Reduction, 1st edn. Springer, New York, https://doi.org/10.1007/978-0-387-39351-3

Liu J, Han J (2014) Spectral clustering. In: Aggarwal CC, Reddy CK (eds) Data Clustering, 1st edn. Chapman and Hall/CRC, Boca Raton, p 177–200, https://doi.org/10.1201/9781315373515

Lloyd S (1982) Least squares quantization in PCM. IEEE Trans Inf Theory 28(2):129–137. https://doi.org/10.1109/TIT.1982.1056489

Ma Y, Fu Y (eds) (2012) Manifold Learning Theory and Applications, vol 434, 1st edn. CRC press, Boca Raton, https://doi.org/10.1201/b11431

van der Maaten L, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(86):2579–2605. URL http://jmlr.org/papers/v9/vandermaaten08a.html

Mautz D, Ye W, Plant C, et al (2017) Towards an optimal subspace for k-means. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 365–373, https://doi.org/10.1145/3097983.3097989

McInnes L (2018) Using UMAP for Clustering. URL https://umap-learn.readthedocs.io/en/latest/clustering.html, [Online; accessed 11-January-2022]

McInnes L, Healy J, Melville J (2018) Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint URL https://arxiv.org/abs/1802.03426

Mittal M, Goyal LM, Hemanth DJ, et al (2019) Clustering approaches for high-dimensional databases: A review. WIREs Data Min Knowl Discov 9(3):e1300. https://doi.org/10.1002/widm.1300

Mu Z, Wu Y, Yin H, et al (2020) Study on single-phase ground fault location of distribution network based on MDS and DBSCAN clustering. In: 2020 39th Chinese Control Conference (CCC), IEEE, pp 6146–6150, https://doi.org/10.23919/CCC50068.2020.9188678

Mukherjee S, Asnani H, Lin E, et al (2019) ClusterGAN: Latent space clustering in generative adversarial networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, pp 4610–4617, https://doi.org/10.1609/aaai.v33i01.33014610

Nane S, Nayar S, Murase H (1996) Columbia object image library: COIL-20. Tech. rep., Department of Computer Science, Columbia University, New York

Niyogi P, Smale S, Weinberger S (2011) A topological view of unsupervised learning from noisy data. SIAM J Comput 40(3):646–663. https://doi.org/10.1137/090762932

Pandove D, Goel S, Rani R (2018) Systematic review of clustering high-dimensional and large datasets. ACM Trans Knowl Discov Data 12(2):1–68.

https://doi.org/10.1145/3132088

Pealat C, Bouleux G, Cheutet V (2021) Improved time-series clustering with UMAP dimension reduction method. In: 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, pp 5658–5665, https://doi.org/10.1109/ICPR48806.2021.9412261

Putri GH, Read MN, Koprinska I, et al (2019) Dimensionality reduction for clustering and cluster tracking of cytometry data. In: Tetko IV, Kůrková V, Karpov P, et al (eds) Artificial Neural Networks and Machine Learning ICANN 2019: Text and Time Series. ICANN 2019. Lecture Notes in Computer Science, vol 11730. Springer, Cham, pp 624–640, https://doi.org/10.1007/978-3-030-30490-4_50

Rasmussen C (2000) The infinite gaussian mixture model. In: Solla S, Leen T, Müller K (eds) Advances in Neural Information Processing Systems, vol 12. MIT Press, URL https://papers.nips.cc/paper/1999/hash/97d98119037c5b8a9663cb21fb8ebf47-Abstract.html

Saxena A, Prasad M, Gupta A, et al (2017) A review of clustering techniques and developments. Neurocomputing 267:664–681. https://doi.org/10.1016/j.neucom.2017.06.053

Schubert E, Sander J, Ester M, et al (2017) DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. ACM Trans Database Syst 42(3):1–21. https://doi.org/10.1145/3068335

Scitovski R, Sabo K, Martínez Álvarez F, et al (2021) Cluster Analysis and Applications, 1st edn. Springer, Cham, https://doi.org/10.1007/978-3-030-74552-3

Souvenir R, Pless R (2005) Manifold clustering. In: Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, IEEE, pp 648–653, https://doi.org/10.1109/ICCV.2005.149

Tenenbaum JB, De Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323. https://doi.org/10.1126/science.290.5500.2319

Thrun MC, Ultsch A (2020) Clustering benchmark datasets exploiting the fundamental clustering problems. Data Brief 30:105,501. https://doi.org/10.1016/j.dib.2020.105501

Ultsch A (2005) Clustering with SOM: U*C. In: Proc. Workshop on Self-Organizing Maps, Paris, France, https://doi.org/10.13140/RG.2.1.2394.5446

Ultsch A, Lötsch J (2020) The fundamental clustering and projection suite (FCPS): A dataset collection to test the performance of clustering and data projection algorithms. Data 5(1). https://doi.org/10.3390/data5010013

Vinh NX, Epps J, Bailey J (2010) Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. J Mach Learn Res 11(95):2837–2854. URL https://jmlr.org/papers/v11/vinh10a.html

Von Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416. https://doi.org/10.1007/s11222-007-9033-z

Wang Y, Huang H, Rudin C, et al (2021) Understanding how dimension reduction tools work: An empirical approach to deciphering t-SNE, UMAP, TriMap, and PaCMAP for data visualization. J Mach Learn Res 22:1–73. URL http://jmlr.org/papers/v22/20-1061.html

Wasserman L (2018) Topological data analysis. Annu Rev Stat Appl 5(1):501–532. https://doi.org/10.1146/annurev-statistics-031017-100045

Wolf FA, Hamey FK, Plass M, et al (2019) PAGA: graph abstraction reconciles clustering with trajectory inference through a topology preserving map of single cells. Genome Biol 20(59):1–9. https://doi.org/10.1186/s13059-019-1663-x

Xiao H, Rasul K, Vollgraf R (2017) Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. arXiv preprint URL https://arxiv.org/abs/1708.07747

Zimek A, Vreeken J (2015) The blind men and the elephant: on meeting the problem of multiple truths in data from clustering and pattern mining perspectives. Mach Learn 98(1):121–155. https://doi.org/10.1007/s10994-013-5334-y

Part IV.

Conclusion

11. Concluding Remarks and Outlook

> The first principle is that you must not fool yourself — and you are the easiest person to fool.
> — Richard Feynman

11.1. Summary and General Implications

The main goal of this thesis was to work toward more reliability in machine learning research. The focus was on unsupervised manifold learning, yet we also elaborated on benchmark studies as an approach to improve replicability in supervised learning. In general, various existing methods and concepts in these two fields have been extensively reviewed and evaluated, i.e., *tested*, which allowed us to identify strengths and weaknesses and draw some general conceptual conclusions that – we would argue – considerably contribute to an improvement of reliability in machine learning.

In particular, the provided contributions improve conceptual clarity in functional data analysis, outlier detection, cluster analysis, and manifold learning. For example, regarding outlier detection, we provide a framework that distinguishes structural and distributional outliers. This fills a gap in the field because – as outline based on the discussion of Zimek & Filzmoser (2018) – (1) there are two more or less vague notions of outliers ("real" and "apparent" in the terminology of Zimek & Filzmoser (2018)) and (2) there is no principled, underlying conceptualization that allows moving from the vague notions to a precise definition that is capable of reflecting the two types of outliers. We provide such a conceptualization in Chapters 8 and 9. Note that this particularly improves the theoretical understanding of outlier detection in functional data, since there are also imprecise and contradictory definitions of outliers in FDA (e.g., see Arribas-Gil & Romo, 2015). Moreover, it is often claimed that outlier detection in high-dimensional data is problematic, if not impossible. Chapters 8 and 9 demonstrate that high-dimensional settings need not be more challenging than low-dimensional settings.

However, this implies that the intrinsic structure of the data set is not fully reflected by a single, connected manifold, as is assumed in the standard notion of manifold learning. There the goal is to infer and reflect the structure of this single manifold, i.e., isometrically retaining the *inner geometry* of a data set. In other settings, however, this assumption appears inappropriate. For example, in outlier detection, we specify two different manifolds that allow us to reflect the two different outlier types. It is crucial to explicitly consider these manifolds to be subsets of an (metric) ambient space that induces a notion of distance between the manifolds. We call this the *outer geometry* of a data set that is inherited from the observation space defined by the variables/features. In cluster analysis, on the other hand, we show that preserving (parts of) the *topological* structure of the data set, specifically: its unconnected components, is the most important aspect, not inferring or preserving its metric structure as expressed in its *inner* or *outer geometry*.

In summary, from a general conceptual viewpoint, this thesis argues that:

1. The standard manifold assumption of a single, connected manifold is not appropriate for exploratory data analysis tasks *outlier detection* and *cluster analysis*.
2. There are two types of outliers, *structural* and *distributional*, that can be precisely specified based on a more general notion of manifold learning assuming at least two distinct data manifolds in a shared ambient space.
3. Cluster detection can be considerably enhanced if one focuses on leveraging unconnected components, i.e., topological features, of a data set. Settings where clusters are allowed to overlap should be more strictly distinguished from settings where clusters are clearly separated.
4. These aspects correspond to the *inner geometry*, *outer geometry*, and the *topology* of a data set. In particular, we would argue these terms much better reflect the discussed aspects than the notions of *local* and *global* structure preservation that manifold learning methods are usually evaluated on and distinguished by.

These insights improve conceptual clarity and theoretical understanding but also have crucial practical implications. Equipped with these problem-adequate and usefully clear conceptualizations, we can employ suitably adapted simple and well-established methods in a variety of situations that are often tackled with highly complex and/or specialized approaches.
For example, we have demonstrated that the simple approach of using MDS and LOF for outlier detection yields performances comparable to those of methodologically complex and functional-data-specific methods. At the same time, the approach naturally extends to other non-tabular and high-dimensional data types such as images and graphs. Furthermore, the simple approach of combining DBSCAN and UMAP can lead to better performances than specialized (deep-learning based) methods for cluster detection on regularly used benchmark data (without the need to specify the number of clusters). In all cases, very little hyperparameter tuning was necessary, a crucial aspect in unsupervised settings where there is so far no established procedure to reliably select a method's hyperparameters.
In essence, the thesis contributes to a better understanding of possible structures in data from a general conceptual perspective. These structures (*unconnected components*, *structural* and *distributional outliers*, *connected (nonlinear) manifolds*) can be (jointly) present in a data set in some situations but not in others. Since the methods vary in their ability to detect certain structures, it may not be possible to simultaneously infer and reflect all structures present in a data set optimally. This is an important aspect to keep in mind in applications and it should be very clearly specified which structures are to be inferred. The present thesis provides concepts and terminology to do so.

11.2. Future Directions

In general, concepts from (topological) manifold learning and topological data analysis gain increasing attention in methodological research and application. For example, they are used to improve and explain supervised, in particular, deep learning methods and to inform about data in specific applications (e.g., see Birdal et al., 2021; Gardner et al., 2022; Gong et al., 2019; Guan et al., 2020; Kim et al., 2020; Ross et al., 2022). In addition to these important research directions, we would like to point out two directions we consider particularly relevant given the insights of the thesis.
First of all, our contributions to outlier detection and cluster analysis suggest that the procedure of using label information to evaluate and compare unsupervised methods should be

treated with great caution. This is not a new insight, but with the concepts of *distributional* in contrast to *structural* outliers and the *connected components* of a data set in contrast to a data set's *classes induced by labels*, it is possible to more clearly specify and evaluate the problem. For example, it would be interesting to investigate how consistent the *connected components* and the *classes induced by labels* of frequently used benchmark data sets are (cf. Guan & Loew, 2021). In outlier detection, methods could be compared based on data sets with different outlier structures: distributional and structural, only distributional or only structural, different forms of structural outliers versus a single form, etc.
Finally, drawing a connection between the two distinct contributions of the study, it would be interesting to investigate how manifold learning can inform benchmark studies. For example, researchers could use (topological) manifold learning to infer the intrinsic structures of the considered data sets to assess which data sets are related in terms of internal characteristics. Even more importantly, it would be of great interest to investigate whether it is possible to use the different concepts of intrinsic data structures to more precisely define the *DGP populations* real data benchmark studies intend to generalize to.

References

Ackerman, M., Ben-David, S., & Loker, D. (2010). Characterization of Linkage-based Clustering. *COLT*, 270–281.

Agrawal, A., Ali, A., & Boyd, S. (2021). Minimum-Distortion Embedding. *Foundations and Trends® in Machine Learning*, *14*(3), 211–378. https://doi.org/10.1561/2200000090

Alaız, C. M. (2015). Diffusion Maps Parameters Selection Based on Neighbourhood Preservation. *Computational Intelligence*, 6.

Allaoui, M., Kherfi, M. L., & Cheriet, A. (2020). Considerably Improving Clustering Algorithms Using UMAP Dimensionality Reduction Technique: A Comparative Study. In A. El Moataz, D. Mammass, A. Mansouri, & F. Nouboud (Eds.), *Image and Signal Processing* (pp. 317–325). Springer International Publishing. https://doi.org/10.1007/978-3-030-51935-3_34

Alpaydin, E. (1999). Combined 5 × 2 cv F Test for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, *11*(8), 1885–1892. https://doi.org/10.1162/089976699300016007

Arnold, B., Bowler, L., Gibson, S., Herterich, P., Higman, R., Krystalli, A., Morley, A., O'Reilly, M., Whitaker, K., & The Turing Way Community. (2019). *The Turing Way: A Handbook for Reproducible Data Science*. Zenodo. https://doi.org/10.5281/zenodo.3233986

Arribas-Gil, A., & Romo, J. (2015). Discussion of "Multivariate functional outlier detection." *Statistical Methods & Applications*, *24*(2), 263–267. https://doi.org/10.1007/s10260-015-0328-5

Aßenmacher, M., & Heumann, C. (2020). On the comparability of pre-trained language models. *Proceedings of the 5th Swiss Text Analytics Conference (SwissText) & 16th Conference on Natural Language Processing (KONVENS)*. http://ceur-ws.org/Vol-2624/paper2.pdf

Aßenmacher, M., Schulze, P., & Heumann, C. (2021). Benchmarking down-scaled (not so large) pre-trained language models. *Proceedings of the 17th Conference on Natural Language Processing (KONVENS 2021)*, 1427. https://aclanthology.org/2021.konvens-1.2

Bagnall, A., Lines, J., Bostrom, A., Large, J., & Keogh, E. (2017). The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, *31*(3), 606–660. https://doi.org/10.1007/s10618-016-0483-9

Baker, M. (2016). 1,500 scientists lift the lid on reproducibility. *Nature*, *533*(7604), 452–454. https://doi.org/10.1038/533452a

Barba, L. A. (2018). *Terminologies for Reproducible Research. arXiv:1802.03311*. http://arxiv.org/abs/1802.03311

Bates, S., Hastie, T., & Tibshirani, R. (2021). *Cross-validation: What does it estimate and how well does it do it? arXiv:2104.00673*. https://doi.org/10.48550/arXiv.2104.00673

Becht, E., McInnes, L., Healy, J., Dutertre, C.-A., Kwok, I. W. H., Ng, L. G., Ginhoux, F., & Newell, E. W. (2019). Dimensionality reduction for visualizing single-cell data using UMAP. *Nature Biotechnology*, *37*(1), 38–44. https://doi.org/10.1038/nbt.4314

Belkin, M., & Niyogi, P. (2003). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, *15*(6), 1373–1396. https://doi.org/10.1162/089976603321780317

Belkina, A. C., Ciccolella, C. O., Anno, R., Halpert, R., Spidlen, J., & Snyder-Cappione, J. E.

(2019). Automated optimized parameters for T-distributed stochastic neighbor embedding improve visualization and analysis of large datasets. *Nature Communications*, *10*(1), 1–12. https://doi.org/10.1038/s41467-019-13055-y

Ben-David, S., & Ackerman, M. (2008). Measures of Clustering Quality: A Working Set of Axioms for Clustering. In D. Koller, D. Schuurmans, Y. Bengio, & L. Bottou (Eds.), *Advances in Neural Information Processing Systems* (Vol. 21). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2008/hash/beed13602b9b0e6ecb5b568ff5058f07-Abstract.html

Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *35*(8), 1798–1828. https://doi.org/10.1109/TPAMI.2013.50

Bengio, Y., & Grandvalet, Y. (2004). No Unbiased Estimator of the Variance of K-Fold Cross-Validation. *Journal of Machine Learning Research*, *5*, 1089–1105. https://www.jmlr.org/papers/volume5/grandvalet04a/grandvalet04a.pdf

Birdal, T., Lou, A., Guibas, L. J., & Simsekli, U. (2021). Intrinsic Dimension, Persistent Homology and Generalization in Neural Networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, & J. W. Vaughan (Eds.), *Advances in Neural Information Processing Systems* (Vol. 34, pp. 6776–6789). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2021/hash/35a12c43227f217207d4e06ffefe39d3-Abstract.html

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A.-L., Deng, D., & Lindauer, M. (2021). *Hyperparameter Optimization: Foundations, Algorithms, Best Practices and Open Challenges. arXiv:2107.05847.* https://doi.org/10.48550/arXiv.2107.05847

Bischl, B., Mersmann, O., Trautmann, H., & Weihs, C. (2012). Resampling Methods for Meta-Model Validation with Recommendations for Evolutionary Computation. *Evolutionary Computation*, *20*(2), 249–275. https://doi.org/10.1162/EVCO_a_00069

Bojchevski, A., Shchur, O., Zügner, D., & Günnemann, S. (2018). NetGAN: Generating Graphs via Random Walks. *Proceedings of the 35th International Conference on Machine Learning*, 610–619. https://proceedings.mlr.press/v80/bojchevski18a.html

Bouckaert, R. R. (2004). Estimating replicability of classifier learning experiments. *Proceedings of the Twenty-First International Conference on Machine Learning*, 15. https://doi.org/10.1145/1015330.1015338

Bouckaert, R. R., & Frank, E. (2004). Evaluating the Replicability of Significance Tests for Comparing Learning Algorithms. In H. Dai, R. Srikant, & C. Zhang (Eds.), *Advances in Knowledge Discovery and Data Mining* (pp. 3–12). Springer. https://doi.org/10.1007/978-3-540-24775-3_3

Boulesteix, A.-L. (2010). Over-optimism in bioinformatics research. *Bioinformatics*, *26*(3), 437–439. https://doi.org/10.1093/bioinformatics/btp648

Boulesteix, A.-L. (2013). On representative and illustrative comparisons with real data in bioinformatics: Response to the letter to the editor by Smith et al. *Bioinformatics*, *29*(20), 2664–2666. https://doi.org/10.1093/bioinformatics/btt458

Boulesteix, A.-L., Hable, R., Lauer, S., & Eugster, M. J. A. (2015). A Statistical Framework for Hypothesis Testing in Real Data Comparison Studies. *The American Statistician*, *69*(3), 201–212. https://doi.org/10.1080/00031305.2015.1005128

Boulesteix, A.-L., Hoffmann, S., Charlton, A., & Seibold, H. (2020). A replication crisis in methodological research? *Significance*, *17*(5), 18–21. https://doi.org/10.1111/1740-9713.01444

Boulesteix, A.-L., Lauer, S., & Eugster, M. J. A. (2013). A Plea for Neutral Comparison Studies in Computational Sciences. *PLoS ONE*, *8*(4), e61562. https://doi.org/10.1371/journal.pone.0061562

Boulesteix, A.-L., Wilson, R., & Hapfelmeier, A. (2017). Towards evidence-based computational statistics: Lessons from clinical research on the role and design of real-data benchmark studies. *BMC Medical Research Methodology*, *17*(1). https://doi.org/10.1186/s12874-017-0417-2

Bubenik, P. (2015). Statistical topological data analysis using persistence landscapes. *Journal of Machine Learning Research*, *16*(3), 77–102. http://jmlr.org/papers/v16/bubenik15a.html

Buchka, S., Hapfelmeier, A., Gardner, P. P., Wilson, R., & Boulesteix, A.-L. (2021). On the optimistic performance evaluation of newly introduced bioinformatic methods. *Genome Biology*, *22*(1), 152. https://doi.org/10.1186/s13059-021-02365-4

Campos, G. O., Zimek, A., Sander, J., Campello, R. J. G. B., Micenková, B., Schubert, E., Assent, I., & Houle, M. E. (2016). On the evaluation of unsupervised outlier detection: Measures, datasets, and an empirical study. *Data Mining and Knowledge Discovery*, *30*(4), 891–927. https://doi.org/10.1007/s10618-015-0444-8

Cantini, L., Zakeri, P., Hernandez, C., Naldi, A., Thieffry, D., Remy, E., & Baudot, A. (2021). Benchmarking joint multi-omics dimensionality reduction approaches for the study of cancer. *Nature Communications*, *12*(1), 124. https://doi.org/10.1038/s41467-020-20430-7

Carlsson, G., & Mémoli, F. (2013). Classifying Clustering Schemes. *Foundations of Computational Mathematics*, *13*(2), 221–252. https://doi.org/10.1007/s10208-012-9141-9

Cayton, L. (2005). *Algorithms for manifold learning* (No. 1-17; Vol. 12). Univ. of California at San Diego Tech. Rep.

Chazal, F., & Michel, B. (2021). An Introduction to Topological Data Analysis: Fundamental and Practical Aspects for Data Scientists. *Frontiers in Artificial Intelligence*, *4*. https://www.frontiersin.org/articles/10.3389/frai.2021.667963

Chen, D., & Müller, H.-G. (2012). Nonlinear manifold representations for functional data. *Annals of Statistics*, *40*(1), 1–29. https://doi.org/10.1214/11-AOS936

Chen, L., & Buja, A. (2009). Local Multidimensional Scaling for Nonlinear Dimension Reduction, Graph Drawing, and Proximity Analysis. *Journal of the American Statistical Association*, *104*(485), 209–219. https://doi.org/10.1198/jasa.2009.0111

Clémençon, S., & Jakubowicz, J. (2013). Scoring anomalies: A M-estimation formulation. *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, 659–667. https://proceedings.mlr.press/v31/clemencon13a.html

Coifman, R. R., & Lafon, S. (2006). Diffusion maps. *Applied and Computational Harmonic Analysis*, *21*(1), 5–30. https://doi.org/10.1016/j.acha.2006.04.006

Cox, M. A. A., & Cox, T. F. (2008). Multidimensional Scaling. In C. Chen, W. Härdle, & A. Unwin (Eds.), *Handbook of Data Visualization* (pp. 315–347). Springer. https://doi.org/10.1007/978-3-540-33037-0_14

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., Hormozdiari, F., Houlsby, N., Hou, S., Jerfel, G., Karthikesalingam, A., Lucic, M., Ma, Y., McLean, C., Mincu, D., … Sculley, D. (2020). Underspecification Presents Challenges for Credibility in Modern Machine Learning. *arXiv:2011.03395 [Cs, Stat]*. http://arxiv.org/abs/2011.03395

Dehghani, M., Tay, Y., Gritsenko, A. A., Zhao, Z., Houlsby, N., Diaz, F., Metzler, D., & Vinyals, O. (2021). The Benchmark Lottery. *arXiv:2107.07002 [Cs]*. http://arxiv.org/abs/2107.07002

Demšar, J. (2006). Statistical Comparisons of Classifiers over Multiple Data Sets. *Journal of Machine Learning Research*, *7*, 1–30. https://www.jmlr.org/papers/volume7/demsar06a/demsar06a.pdf

Dietterich, T. G. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, *10*(7), 1895–1923. https://doi.org/10.1162/08

9976698300017197

Dimeglio, C., Gallón, S., Loubes, J.-M., & Maza, E. (2014). A robust algorithm for template curve estimation based on manifold embedding. *Computational Statistics & Data Analysis*, *70*, 373–386. https://doi.org/10.1016/j.csda.2013.09.030

Domingues, R., Filippone, M., Michiardi, P., & Zouaoui, J. (2018). A comparative evaluation of outlier detection algorithms: Experiments and analyses. *Pattern Recognition*, *74*, 406–421. https://doi.org/10.1016/j.patcog.2017.09.037

Drummond, C. (2009, June). Replicability is not Reproducibility: Nor is it Good Science. *Proceedings of the Evaluation Methods for Machine Learning Workshop at the 26th ICML*. https://www.site.uottawa.ca/~cdrummon/pubs/ICMLws09.pdf

Eisinga, R., Heskes, T., Pelzer, B., & Te Grotenhuis, M. (2017). Exact p-values for pairwise comparison of Friedman rank sums, with application to comparing classifiers. *BMC Bioinformatics*, *18*(1), 68. https://doi.org/10.1186/s12859-017-1486-2

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, *96*, 226–231.

Eugster, M. J. A., Hothorn, T., & Leisch, F. (2012). Domain-Based Benchmark Experiments: Exploratory and Inferential Analysis. *Austrian Journal of Statistics*, *41*(1), 5–26. https://doi.org/10.17713/ajs.v41i1.185

Ferraty, F., & Vieu, P. (2006). *Nonparametric Functional Data Analysis: Theory and Practice* (1st ed.). New York: Springer. https://doi.org/10.1007/0-387-36620-2

Feynman, R. P. (1974). *Cargo Cult Science*. https://calteches.library.caltech.edu/51/2/CargoCult.htm

Frambach, J. M., Vleuten, C. P. van der, & Durning, S. J. (2013). AM Last Page: Quality Criteria in Qualitative and Quantitative Research. *Academic Medicine*, *88*(4), 552.

Gan, J., & Tao, Y. (2015). DBSCAN Revisited: Mis-Claim, Un-Fixability, and Approximation. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 519–530. https://doi.org/10.1145/2723372.2737792

Gardner, R. J., Hermansen, E., Pachitariu, M., Burak, Y., Baas, N. A., Dunn, B. A., Moser, M.-B., & Moser, E. I. (2022). Toroidal topology of population activity in grid cells. *Nature*, *602*(7895), 123–128. https://doi.org/10.1038/s41586-021-04268-7

Gencoglu, O., van Gils, M., Guldogan, E., Morikawa, C., Süzen, M., Gruber, M., Leinonen, J., & Huttunen, H. (2019). HARK Side of Deep Learning – From Grad Student Descent to Automated Machine Learning. *arXiv:1904.07633 [Cs]*. http://arxiv.org/abs/1904.07633

Ghahramani, Z. (2004). Unsupervised Learning. In O. Bousquet, U. von Luxburg, & G. Rätsch (Eds.), *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2 - 14, 2003, Tübingen, Germany, August 4 - 16, 2003, Revised Lectures* (pp. 72–112). Springer. https://doi.org/10.1007/978-3-540-28650-9_5

Gisbrecht, A., & Hammer, B. (2015). Data visualization by nonlinear dimensionality reduction. *WIREs Data Mining and Knowledge Discovery*, *5*(2), 51–73. https://doi.org/10.1002/widm.1147

Goix, N. (2016). How to Evaluate the Quality of Unsupervised Anomaly Detection Algorithms? *arXiv:1607.01152 [Cs, Stat]*. http://arxiv.org/abs/1607.01152

Goldstein, M., & Uchida, S. (2016). A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. *PLOS ONE*, *11*(4), e0152173. https://doi.org/10.1371/journal.pone.0152173

Gong, S., Boddeti, V. N., & Jain, A. K. (2019). On the Intrinsic Dimensionality of Image Representations. *arXiv:1803.09672 [Cs, Stat]*. http://arxiv.org/abs/1803.09672

Guan, S., & Loew, M. (2021). A Novel Intrinsic Measure of Data Separability. *arXiv:2109.05180 [Cs, Math, Stat]*. http://arxiv.org/abs/2109.05180

Guan, S., Loew, M., & Ko, H. (2020). *Data Separability for Neural Network Classifiers and the Development of a Separability Index* (No. arXiv:2005.13120). arXiv. https://doi.org/10.48550/arXiv.2005.13120

Guyon, I., Saffari, A., Dror, G., & Cawley, G. (2010). Model Selection: Beyond the Bayesian/Frequentist Divide. *Journal of Machine Learning Research*, *11*(3), 61–87. http://jmlr.org/papers/v11/guyon10a.html

Hand, D. J. (2006). Classifier Technology and the Illusion of Progress. *Statistical Science*, *21*(1), 1–14. https://doi.org/10.1214/088342306000000060

Happ, C., & Greven, S. (2018). Multivariate Functional Principal Component Analysis for Data Observed on Different (Dimensional) Domains. *Journal of the American Statistical Association*, *113*(522), 649–659. https://doi.org/10.1080/01621459.2016.1273115

Happ, C., Scheipl, F., Gabriel, A.-A., & Greven, S. (2019). A general framework for multivariate functional principal component analysis of amplitude and phase variation. *Stat*, *8*(1), e220. 10.1002/sta4.220

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). New York: Springer. https://doi.org/10.1007/978-0-387-84858-7

Heil, B. J., Hoffman, M. M., Markowetz, F., Lee, S.-I., Greene, C. S., & Hicks, S. C. (2021). Reproducibility standards for machine learning in the life sciences. *Nature Methods*, *18*(10), 1132–1135. https://doi.org/10.1038/s41592-021-01256-7

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., & Meger, D. (2018). Deep Reinforcement Learning That Matters. *Proceedings of the AAAI Conference on Artificial Intelligence*, *32*(1). https://ojs.aaai.org/index.php/AAAI/article/view/11694

Hothorn, T., Leisch, F., Zeileis, A., & Hornik, K. (2005). The Design and Analysis of Benchmark Experiments. *Journal of Computational and Graphical Statistics*, *14*(3), 675–699. https://doi.org/10.1198/106186005X59630

Hutson, M. (2018). Artificial intelligence faces reproducibility crisis. *Science*, *359*(6377), 725–726. https://doi.org/10.1126/science.359.6377.725

Ioannidis, J. P. A. (2005). Why Most Published Research Findings Are False. *PLOS Medicine*, *2*(8), e124. https://doi.org/10.1371/journal.pmed.0020124

Jelizarow, M., Guillemot, V., Tenenhaus, A., Strimmer, K., & Boulesteix, A.-L. (2010). Over-optimism in bioinformatics: An illustration. *Bioinformatics*, *26*(16), 1990–1998. https://doi.org/10.1093/bioinformatics/btq323

Kim, K., Kim, J., Zaheer, M., Kim, J., Chazal, F., & Wasserman, L. (2020). PLLay: Efficient Topological Layer based on Persistent Landscapes. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems* (Vol. 33, pp. 15965–15977). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2020/file/b803a9254688e259cde2ec0361c8abe4-Paper.pdf

Kleinberg, J. (2002). An Impossibility Theorem for Clustering. In S. Becker, S. Thrun, & K. Obermayer (Eds.), *Advances in Neural Information Processing Systems* (Vol. 15). MIT Press. https://proceedings.neurips.cc/paper/2002/hash/43e4e6a6f341e00671e123714de019a8-Abstract.html

Klemenjak, C., Makonin, S., & Elmenreich, W. (2020). Towards Comparability in Non-Intrusive Load Monitoring: On Data and Performance Evaluation. *2020 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 1–5. https://doi.org/10.1109/ISGT45199.2020.9087706

Kobak, D., & Berens, P. (2019). The art of using t-SNE for single-cell transcriptomics. *Nature Communications*, *10*(1), 5416. https://doi.org/10.1038/s41467-019-13056-x

Kobak, D., & Linderman, G. C. (2021). Initialization is critical for preserving global data structure in both t-SNE and UMAP. *Nature Biotechnology*, *39*(2), 156–157. https://doi.or

g/10.1038/s41587-020-00809-z

Kraemer, G., Reichstein, M., & Mahecha, M., D. (2018). dimRed and coRanking - Unifying Dimensionality Reduction in R. *The R Journal*, *10*(1), 342. https://doi.org/10.32614/RJ-2018-039

Lee, J. A., & Verleysen, M. (2009). Quality assessment of dimensionality reduction: Rank-based criteria. *Neurocomputing*, *72*(7-9), 1431–1443. https://doi.org/10.1016/j.neucom.2008.12.017

Lee, J. A., & Verleysen, M. (2007). *Nonlinear Dimensionality Reduction* (1st ed.). New York: Springer. 10.1007/978-0-387-39351-3

Lee, J. A., & Verleysen, M. (2008). Quality assessment of nonlinear dimensionality reduction based on K-ary neighborhoods. In Y. Saeys, H. Liu, I. Inza, L. Wehenkel, & Y. V. de Pee (Eds.), *Proceedings of the workshop on new challenges for feature selection in data mining and knowledge discovery at ECML/PKDD 2008* (Vol. 4, pp. 21–35). PMLR. https://proceedings.mlr.press/v4/lee08a.html

Li, C., Dakkak, A., Xiong, J., & Hwu, W. (2019). Challenges and Pitfalls of Machine Learning Evaluation and Benchmarking. *arXiv:1904.12437 [Cs]*. http://arxiv.org/abs/1904.12437

Liang, J., Chenouri, S., & Small, C. G. (2020). A new method for performance analysis in nonlinear dimensionality reduction. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, *13*(1), 98–108. https://doi.org/10.1002/sam.11445

Linderman, G. C., & Steinerberger, S. (2019). Clustering with t-SNE, Provably. *SIAM Journal on Mathematics of Data Science*, *1*(2), 313–332. https://doi.org/10.1137/18M1216134

Lucic, M., Kurach, K., Michalski, M., Gelly, S., & Bousquet, O. (2018). Are GANs Created Equal? A Large-Scale Study. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 31). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2018/hash/e46de7e1bcaaced9a54f1e9d0d2f800d-Abstract.html

Lueks, W., Mokbel, B., Biehl, M., & Hammer, B. (2011). *How to Evaluate Dimensionality Reduction? - Improving the Co-ranking Matrix*. arXiv. http://arxiv.org/abs/1110.3917

Luxburg, U. von, Williamson, R. C., & Guyon, I. (2012). Clustering: Science or Art? *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, 65–79. https://proceedings.mlr.press/v27/luxburg12a.html

Ma, Y., & Fu, Y. (Eds.). (2011). *Manifold Learning Theory and Applications* (1st ed.). Boca Raton: CRC Press. https://doi.org/10.1201/b11431

Maaten, L. van der, & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(Nov), 2579–2605. http://www.jmlr.org/papers/v9/vandermaaten08a.html

Marie, B., Fujita, A., & Rubino, R. (2021). Scientific Credibility of Machine Translation Research: A Meta-Evaluation of 769 Papers. *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 7297–7306. https://doi.org/10.18653/v1/2021.acl-long.566

Marques, H. O., Campello, R. J., Sander, J., & Zimek, A. (2020). Internal Evaluation of Unsupervised Outlier Detection. *ACM Transactions on Knowledge Discovery from Data (TKDD)*. https://doi.org/10.1145/3394053

McInnes, L. (2018). *How UMAP Works*. [Online; accessed 29-July-2022]. https://umap-learn.readthedocs.io/en/latest/how_umap_works.html

McInnes, L., Healy, J., & Melville, J. (2020). UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv:1802.03426 [Cs, Stat]*. http://arxiv.org/abs/1802.03426

Mead, A. (1992). Review of the Development of Multidimensional Scaling Methods. *Journal*

of the Royal Statistical Society: Series D (The Statistician), *41*(1), 27–39. https://doi.org/10.2307/2348634

Munafò, M. R., Nosek, B. A., Bishop, D. V. M., Button, K. S., Chambers, C. D., Percie du Sert, N., Simonsohn, U., Wagenmakers, E.-J., Ware, J. J., & Ioannidis, J. P. A. (2017). A manifesto for reproducible science. *Nature Human Behaviour*, *1*(1), 1–9. https://doi.org/10.1038/s41562-016-0021

Myrtveit, I., Stensrud, E., & Shepperd, M. (2005). Reliability and validity in comparative studies of software prediction models. *IEEE Transactions on Software Engineering*, *31*(5), 380–391. https://doi.org/10.1109/TSE.2005.58

Nadeau, C., & Bengio, Y. (2003). Inference for the Generalization Error. *Machine Learning*, *3*(52), 239–281. https://doi.org/10.1023/A:1024068626366

Niyogi, P., Smale, S., & Weinberger, S. (2011). A Topological View of Unsupervised Learning from Noisy Data. *SIAM Journal on Computing*, *40*, 646–663. https://doi.org/10.1137/090762932

Pineau, J., Vincent-Lamarre, P., Sinha, K., Larivière, V., Beygelzimer, A., d'Alché-Buc, F., Fox, E., & Larochelle, H. (2020). Improving Reproducibility in Machine Learning Research (A Report from the NeurIPS 2019 Reproducibility Program). *arXiv:2003.12206 [Cs, Stat]*. http://arxiv.org/abs/2003.12206

Plesser, H. E. (2018). Reproducibility vs. Replicability: A Brief History of a Confused Terminology. *Frontiers in Neuroinformatics*, *11*(76). https://doi.org/10.3389/fninf.2017.00076

Raff, E. (2019). A step toward quantifying independently reproducible machine learning research. *arXiv:1909.06674 [Cs, Stat]*. http://arxiv.org/abs/1909.06674

Rajagopal, R., & Ranganathan, V. (2017). Evaluation of effect of unsupervised dimensionality reduction techniques on automated arrhythmia classification. *Biomedical Signal Processing and Control*, *34*, 1–8. https://doi.org/10.1016/j.bspc.2016.12.017

Ramsay, J. O., & Silverman, B. W. (2005). *Functional data analysis* (2nd ed.). New York: Springer. https://doi.org/10.1007/b98888

Rand, W. M. (1971). Objective Criteria for the Evaluation of Clustering Methods. *Journal of the American Statistical Association*, *66*(336), 846–850. https://doi.org/10.1080/01621459.1971.10482356

Rendón, E., Abundez, I., Arizmendi, A., & Quiroz, E. M. (2011). Internal versus External cluster validation indexes. *International Journal of Computers Communications & Control*, *5*(1), 27–34.

Rendsburg, L., Heidrich, H., & Luxburg, U. V. (2020). NetGAN without GAN: From Random Walks to Low-Rank Approximations. *Proceedings of the 37th International Conference on Machine Learning*, 8073–8082. https://proceedings.mlr.press/v119/rendsburg20a.html

Rieck, B., & Leitte, H. (2015). Persistent Homology for the Evaluation of Dimensionality Reduction Schemes. *Computer Graphics Forum*, *34*(3), 431–440. https://doi.org/10.1111/cgf.12655

Roberts, P., & Priest, H. (2006). Reliability and validity in research. *Nursing Standard*, *20*(44), 41–46.

Ross, B. L., Loaiza-Ganem, G., Caterini, A. L., & Cresswell, J. C. (2022). *Neural Implicit Manifold Learning for Topology-Aware Generative Modelling* (No. arXiv:2206.11267). arXiv. https://doi.org/10.48550/arXiv.2206.11267

Roweis, S. T., & Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, *290*(5500), 2323–2326. https://doi.org/10.1126/science.290.5500.2323

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., & Xu, X. (2017). DBSCAN Revisited, Revisited: Why and How You Should (Still) Use DBSCAN. *ACM Transactions on Database*

Systems, *42*(3), 1–21. https://doi.org/10.1145/3068335

Schubert, E., Wojdanowski, R., Zimek, A., & Kriegel, H.-P. (2012). On Evaluation of Outlier Rankings and Outlier Scores. In *Proceedings of the 2012 SIAM International Conference on Data Mining (SDM)* (pp. 1047–1058). Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9781611972825.90

Scott, C. D., & Nowak, R. D. (2006). Learning Minimum Volume Sets. *Journal of Machine Learning Research*, *7*(24), 665–704. http://jmlr.org/papers/v7/scott06a.html

Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). Hidden Technical Debt in Machine Learning Systems. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 28). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2015/hash/86df7dcfd896fcaf2674f757a2463eba-Abstract.html

Segebarth, D., Griebel, M., Stein, N., von Collenberg, C. R., Martin, C., Fiedler, D., Comeras, L. B., Sah, A., Schoeffler, V., Lüffe, T., Dürr, A., Gupta, R., Sasi, M., Lillesaar, C., Lange, M. D., Tasan, R. O., Singewald, N., Pape, H.-C., Flath, C. M., & Blum, R. (2020). On the objectivity, reliability, and validity of deep learning enabled bioimage analyses. *eLife*, *9*, e59780. https://doi.org/10.7554/eLife.59780

Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9781107298019

Shang, H. L. (2014). A survey of functional principal component analysis. *AStA Advances in Statistical Analysis*, *98*(2), 121–142. https://doi.org/10.1007/s10182-013-0213-1

Tatman, R., VanderPlas, J., & Dane, S. (2018, June). A Practical Taxonomy of Reproducibility for Machine Learning Research. *Reproducibility in ML Workshop, ICML 2018*. https://openreview.net/forum?id=B1eYYK5QgX

Tenenbaum, J. B., Silva, V. de, & Langford, J. C. (2000). A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, *290*(5500), 2319–2323. https://doi.org/10.1126/science.290.5500.2319

Thomas, A., Feuillard, V., Gramfort, A., & Clémençon, S. (2016). Learning hyperparameters for unsupervised anomaly detection. *Anomaly Detection Workshop, ICML 2016*. https://github.com/albertcthomas/anomaly_tuning

Torgerson, W. S. (1952). Multidimensional scaling: I. Theory and method. *Psychometrika*, *17*(4), 401–419. https://doi.org/10.1007/BF02288916

Ullmann, T., Hennig, C., & Boulesteix, A.-L. (2022). Validation of cluster analysis results on validation data: A systematic framework. *WIREs Data Mining and Knowledge Discovery*, *12*(3), e1444. https://doi.org/10.1002/widm.1444

Unwin, A. (2019). Multivariate Outliers and the O3 Plot. *Journal of Computational and Graphical Statistics*, *28*(3), 635–643. https://doi.org/10.1080/10618600.2019.1575226

Van Mechelen, I., Boulesteix, A.-L., Dangl, R., Dean, N., Guyon, I., Hennig, C., Leisch, F., & Steinley, D. (2018). Benchmarking in cluster analysis: A white paper. *arXiv:1809.10496 [Stat]*. http://arxiv.org/abs/1809.10496

Venna, J., & Kaski, S. (2001). Neighborhood Preservation in Nonlinear Projection Methods: An Experimental Study. In G. Dorffner, H. Bischof, & K. Hornik (Eds.), *Artificial Neural Networks — ICANN 2001* (pp. 485–491). Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-44668-0_68

Vidal, R., Ma, Y., & Sastry, S. S. (2016). *Generalized Principal Component Analysis* (Vol. 40). New York: Springer. https://doi.org/10.1007/978-0-387-87811-9

Vinh, N. X., Epps, J., & Bailey, J. (2010). Information Theoretic Measures for Clusterings Comparison: Variants, Properties, Normalization and Correction for Chance. *Journal of*

Machine Learning Research, *11*(95), 2837–2854. http://jmlr.org/papers/v11/vinh10a.html

Wang, J.-L., Chiou, J.-M., & Müller, H.-G. (2016). Functional Data Analysis. *Annual Review of Statistics and Its Application*, *3*(1), 257–295. https://doi.org/10.1146/annurev-statistics-041715-033624

Wang, Y., Huang, H., Rudin, C., & Shaposhnik, Y. (2021). Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMap, and PaCMAP for Data Visualization. *Journal of Machine Learning Research*, *22*(201), 1–73. http://jmlr.org/papers/v22/20-1061.html

Wasserman, L. (2016). Topological Data Analysis. *arXiv:1609.08227 [Stat]*. http://arxiv.org/abs/1609.08227

Wattenberg, M., Viégas, F., & Johnson, I. (2016). How to Use t-SNE Effectively. *Distill*, *1*(10), e2. https://doi.org/10.23915/distill.00002

Wolpert, D. H. (2020). What is important about the No Free Lunch theorems? *arXiv:2007.10928 [Cs, Stat]*. http://arxiv.org/abs/2007.10928

Young, G., & Householder, A. S. (1938). Discussion of a set of points in terms of their mutual distances. *Psychometrika*, *3*(1), 19–22. https://doi.org/10.1007/BF02287916

Zimek, A., & Filzmoser, P. (2018). There and back again: Outlier detection between statistical reasoning and data mining algorithms. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, *8*(6). https://doi.org/10.1002/widm.1280

Zimmermann, A. (2020). Method evaluation, parameterization, and result validation in unsupervised data mining: A critical survey. *WIREs Data Mining and Knowledge Discovery*, *10*(2), e1330. https://doi.org/10.1002/widm.1330

Zomorodian, A., & Carlsson, G. (2005). Computing Persistent Homology. *Discrete & Computational Geometry*, *33*(2), 249–274. https://doi.org/10.1007/s00454-004-1146-y

Eidesstattliche Versicherung

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

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

München, den 02.08.2022 Moritz Herrmann