Machine Learning in a Setting of Ordinal Distance Information

Dissertation
der Mathematisch-Naturwissenschaftlichen Fakultät
der Eberhard Karls Universität Tübingen
zur Erlangung des Grades eines
Doktors der Naturwissenschaften
(Dr. rer. nat.)

vorgelegt von
Dipl.-Ing. Matthäus Kleindeßner
aus Linz/Österreich

Tübingen
2017

Gedruckt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät der Eberhard Karls Universität Tübingen.

Tag der mündlichen Qualifikation:	21.07.2017
Dekan:	Prof. Dr. Wolfgang Rosenstiel
1. Berichterstatter:	Prof. Dr. Ulrike von Luxburg
2. Berichterstatter:	Prof. Dr. Matthias Hein

Machine Learning in a Setting of Ordinal Distance Information

Matthäus Kleindeßner

Abstract

In this thesis I make some contributions to the development of machine learning in a setting of ordinal distance information. A setting of ordinal distance information or ordinal data for short refers to the following scenario: The objects of interest are elements of a set $\mathcal{X}$ that is equipped with a dissimilarity function $\iota : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, which quantifies how dissimilar two objects are. However, when given a data set $\mathcal{D} \subseteq \mathcal{X}$ for which we want to solve a machine learning problem, we cannot evaluate ι itself. Instead, we only get to see binary answers to some comparisons of dissimilarity values such as

$$\iota(A, B) \overset{?}{<} \iota(C, D),$$

where $A, B, C, D \in \mathcal{D}$. Such a scenario has attracted interest in machine learning in recent years. It is in contrast to a scenario in which ι can directly be evaluated and actual dissimilarity values $\iota(A, B) \in \mathbb{R}$ are observed. The latter scenario is "standard" in machine learning and is referred to as a setting of cardinal distance information.

My contributions are both on the theoretical and on the applied side. After the introductory Chapter 1, I present a result stating the asymptotic uniqueness of ordinal embeddings in Chapter 2. Constructing an ordinal embedding is the main approach to machine learning in a setting of ordinal distance information. The idea is to map the objects of the data set $\mathcal{D}$ to points in a Euclidean space $\mathbb{R}^d$ such that the points preserve the given ordinal data as well as possible (with respect to the Euclidean interpoint distances). I show that for Euclidean data sets, which permit perfect ordinal embeddings, the points are uniquely determined up to a similarity transformation and small individual offsets that uniformly go to zero as the size of the data set goes to infinity. My result is the first of this kind in the literature and proves a long-standing claim dating back to the 1960s. In Chapter 3, I introduce two estimators for the intrinsic dimension of a data set that are based on only ordinal distance information. Although dimensionality estimation is a well-studied problem with a long history, all previous estimators from the literature are based on cardinal distance information. In Chapter 4 and Chapter 5, I provide algorithms for various machine learning problems in a setting of ordinal distance information. My algorithms do not construct an ordinal embedding of the data set, which would mean to transform the given ordinal data back to a "standard" cardinal setting. They rather directly make use of the ordinal data. In doing so, they avoid some of the drawbacks of an ordinal embedding approach. The algorithms that I propose in Chapter 4 are based on estimating the lens depth function or the k-relative neighborhood graph from ordinal distance information and are designed for specific machine learning problems. My algorithms of Chapter 5 yield positive-semidefinite kernel matrices on the data set $\mathcal{D}$ and hence allow to apply any kernel method to $\mathcal{D}$. They are the first generic means for solving machine learning problems in a setting of ordinal distance information in the literature that is different from the ordinal embedding approach.

Zusammenfassung

In dieser Dissertation präsentiere ich Beiträge zur (Weiter-)Entwicklung des maschinellen Lernens in einem Setting ordinaler Abstandsinformation. Ein Setting ordinaler Abstandsinformation oder kurz ordinaler Daten bezeichnet folgendes Szenario: Die Objekte von Interesse sind Elemente einer Menge $\mathcal{X}$, die mit einer Abstandsfunktion $\iota : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ versehen ist, welche die Unähnlichkeit zweier Objekte quantifiziert. Wenn wir ein Problem des maschinellen Lernens für eine Datenmenge $\mathcal{D} \subseteq \mathcal{X}$ lösen wollen, können wir jedoch ι nicht unmittelbar auswerten. Stattdessen beobachten wir lediglich binäre Antworten auf Vergleiche von Abständen wie etwa

$$\iota(A, B) \overset{?}{<} \iota(C, D),$$

wobei $A, B, C, D \in \mathcal{D}$. Ein solches Szenario ist in den letzten Jahren auf zunehmendes Interesse im maschinellen Lernen gestoßen. Es steht einem Szenario, in welchem ι unmittelbar ausgewertet werden kann und tatsächliche Abstandswerte $\iota(A, B) \in \mathbb{R}$ beobachtet werden, gegenüber. Letzteres Szenario ist ein „standardmäßiges“ im maschinellen Lernen und wird als ein Setting kardinaler Abstandsinformation bezeichnet.

Meine Beiträge liegen sowohl im theoretischen als auch im angewandten Bereich. Nach dem einführenden Kapitel 1 präsentiere ich in Kapitel 2 ein Resultat über die asymptotische Eindeutigkeit ordinaler Einbettungen. Eine ordinale Einbettung zu konstruieren ist die übliche Herangehensweise an maschinelles Lernen in einem Setting ordinaler Abstandsinformation. Die Idee dabei ist, die Objekte der Datenmenge $\mathcal{D}$ auf Punkte in einem euklidischen Raum $\mathbb{R}^d$ so abzubilden, dass die Punkte die gegebenen ordinalen Daten bestmöglich widerspiegeln (bezüglich der euklidischen Punktabstände). Ich zeige, dass für euklidische Datenmengen, welche perfekte ordinale Einbettungen erlauben, die Punkte eindeutig bestimmt sind, bis auf eine Ähnlichkeitsabbildung und kleine individuelle Verschiebungen, die gleichmäßig gegen null streben, während die Größe der Datenmenge gegen unendlich strebt. Mein Resultat ist das erste dieser Art in der Literatur und beweist eine seit Langem bestehende Vermutung, die bis in die 1960er-Jahre zurückreicht. In Kapitel 3 führe ich zwei Schätzer für die intrinsische Dimension einer Datenmenge ein, die ausschließlich ordinale Abstandsinformation verwenden. Obwohl Dimensions-Schätzung ein gut untersuchtes Problem mit einer langen Geschichte ist, basieren alle bisherigen Dimensions-Schätzer in der Literatur auf kardinaler Abstandsinformation. In Kapitel 4 und Kapitel 5 präsentiere ich Algorithmen für verschiedene Probleme des maschinellen Lernens in einem Setting ordinaler Abstandsinformation. Meine Algorithmen konstruieren keine ordinale Einbettung der Datenmenge—was bedeuten würde, die gegebenen ordinalen Daten in ein „standardmäßiges“ Setting kardinaler Abstandsinformation zurückzutransformieren. Stattdessen arbeiten sie direkt auf den ordinalen Daten und vermeiden dadurch einige der Nachteile, die mit dem Konstruieren einer ordinalen Einbettung einhergehen. Die Algorithmen, die ich in Kapitel 4 vorschlage, beruhen darauf, die sogenannte Linsen-Tiefenfunktion beziehungsweise den k-relativen Nachbarschaftsgraphen basierend auf ordinaler Abstandsinformation zu schätzen, und lösen spezifische Probleme des maschinellen Lernens. Die Algorithmen aus Kapitel 5 liefern positiv semidefinite Kernmatrizen auf der Datenmenge $\mathcal{D}$ und erlauben daher eine beliebige Kernmethode auf $\mathcal{D}$ anzuwenden. Sie sind die erste allgemeine Methode um Probleme des maschinellen Lernens in einem Setting ordinaler Abstandsinformation zu lösen in der Literatur, die nicht aus dem Konstruieren einer ordinalen Einbettung besteht.

Acknowledgements

I would particularly like to thank my thesis supervisor Ulrike von Luxburg for giving me the opportunity to work in her group, her constant and reliable support all the last four years, and her very pleasant nature. She has been a great help regarding work and beyond.

I would like to thank my colleagues, in particular the two I shared my office with, Sven in Hamburg and Debarghya in Tübingen, but also all the others—Lennard, Morteza, Sascha, Siavash, Tobias, and Yoshikazu—for numerous helpful discussions, the fruitful working atmosphere, and the many joyful things we did together outside of university.

Finally, I would like to thank my family for their everlasting support.

Contents

Chapter 1

Introduction

Assessing similarity between objects is an inherent part of many machine learning problems, be it in an unsupervised task like clustering, in which similar objects should be grouped together, or in a supervised task like classification, where many algorithms are based on the assumption that similar inputs tend to produce similar outputs. In a typical machine learning setting one commonly assumes to be given a data set $\mathcal{D}$ of objects together with a dissimilarity function ι quantifying how "close" objects are to each other. In recent years, however, a whole new branch of the machine learning literature has emerged that relaxes this scenario. Instead of being able to evaluate the dissimilarity function ι itself, we only get to see binary answers to some comparisons of dissimilarity values such as

$$\iota(A,B) \overset{?}{<} \iota(C,D),$$

where $A, B, C, D \in \mathcal{D}$. We refer to any collection of answers to such dissimilarity comparisons as *ordinal distance information* or *ordinal data* as opposed to *cardinal distance information* comprising actual dissimilarity values $\iota(A,B) \in \mathbb{R}$. Note that, in general, the ordinal data is the only knowledge that we have about the data set $\mathcal{D}$. In particular, we do not have any feature representations of the objects whatsoever.

In this thesis we make some contributions to the development of machine learning in a setting of ordinal distance information, both on the theoretical and on the applied side. We will summarize our results in Section 1.5 of this introduction. Before we want to provide some motivation for studying ordinal data in Section 1.1. We make some general assumptions on the dissimilarity function ι as well as introduce some general notation in Section 1.2. In Section 1.3 and Section 1.4, respectively, we review two important aspects of ordinal distance information that are relevant to understand our results: distinguishing ordinal data regarding to which kind of ordinal relationships it is composed of and ordinal embedding. Throughout this introductory chapter we keep the discussion on a somewhat informal level. In subsequent chapters we will be mathematically precise and rigorous.

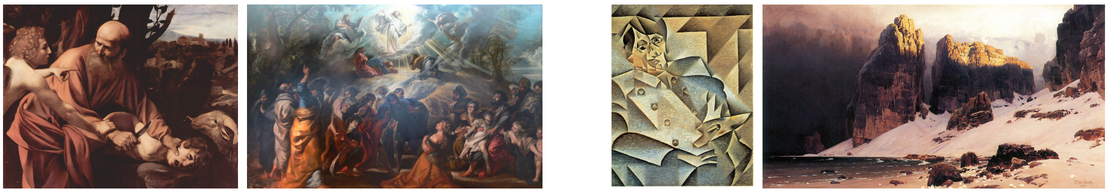


Figure 1.1: One reason for our interest in ordinal data: it is easier for humans to declare that the first two paintings are more similar to each other than the third and the fourth painting are than to provide numerical (dis-)similarity scores.[1]

1.1 Motivation for studying ordinal data

Besides theoretical interest, there are several real-life motivations for studying machine learning tasks in a setting of ordinal distance information:

- Human-based computation / crowdsourcing: In complex tasks, such as estimating the value of a car shown in an image or clustering biographies of celebrities, it can be hard to come up with a meaningful dissimilarity function that can be evaluated automatically, while humans often have a good sense of which objects should be considered (dis-)similar. It is then natural to incorporate the human expertise into the machine learning process. As it is a general phenomenon that humans are significantly better at comparing stimuli than at identifying a single one (Stewart et al., 2005), it is widely believed and accepted that humans are also better and more reliable in assessing dissimilarity on a relative scale ("Painting A is more similar to painting B than painting C is to painting D") than on an absolute one ("The similarity between A and B is 0.8 and the similarity between C and D is 0.3"). This is illustrated by an example in Figure 1.1. For this reason, ordinal questions are often used when humans are involved in gathering distance information. In addition to obtaining more robust results, this also has the advantage that one does not need to align people's different assessment scales.

- There are scenarios in which ordinal distance information is readily available, but the underlying dissimilarity function is completely in the dark. Schultz and Joachims (2003) provide the example of search-engine query logs: if a user clicks on two search results, say A and B, but not on a third result C, then A and B can be assumed to be semantically more similar than A and C, or B and C, are.

- There are several applications where similarity values between objects are computed on a regular basis, but it is clear to the practitioner that these values only reflect a rough picture and should be considered informative only on an ordinal scale level. In this case, providing the numerical similarity scores to a machine learning algorithm can offer the problem that the algorithm interprets them stronger than they are meant to be. For example, discarding the actual values of signal strength measurements and only keeping their order can help to reduce the influence of measurement errors and thus bring some benefit in sensor localization (Liu et al., 2004, Xiao et al., 2006).

[1] The pictures were found on Wikimedia Commons and are in the public domain.

1.2 General assumptions on the dissimilarity function ι and some general notation

By a *dissimilarity function* ι on some set $\mathcal{X}$ we mean a function $\iota : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ measuring dissimilarity between elements of $\mathcal{X}$ in the sense that a higher value of ι means that two elements are more dissimilar—or, equivalently, less similar—to each other. We use the terms *dissimilarity* and *distance* synonymously. The minimal assumptions that we make on ι are the following:

- $\iota(x,y) \geq 0, \quad x,y \in \mathcal{X}$
- $\iota(x,y) = \iota(y,x), \quad x,y \in \mathcal{X}$ (that is ι is symmetric).

Note that we neither assume ι to be positive definite, that is $\iota(x,y) = 0$ if and only if $x = y$, $x,y \in \mathcal{X}$, nor to satisfy the triangle inequality $\iota(x,y) \leq \iota(x,z) + \iota(z,y)$, $x,y,z \in \mathcal{X}$. In particular, $(\mathcal{X}, \iota)$ is not necessarily a metric space. In Chapter 2 and Chapter 3, however, we actually only deal with $\mathcal{X} \subseteq \mathbb{R}^d$ and ι being the *Euclidean metric*, which we also refer to as *Euclidean distance*. The Euclidean metric is induced by the *Euclidean norm*. We write the Euclidean norm as $\|\cdot\|$ or $\|\cdot\|_{\mathbb{R}^d}$ if we want to emphasize the dimension of the space on which it is defined. The corresponding Euclidean inner product is written as $\langle\cdot\,,\cdot\rangle$ or $\langle\cdot\,,\cdot\rangle_{\mathbb{R}^d}$. Recall that, for $a, b \in \mathbb{R}^d$,

$$\|a\|_{\mathbb{R}^d} = \sqrt{\sum_{i=1}^{d} a_i^2}, \qquad \langle a, b\rangle_{\mathbb{R}^d} = \sum_{i=1}^{d} a_i \cdot b_i,$$

and the Euclidean distance between a and b is $\|a-b\|$. If $d = 1$, $\|a-b\|$ equals $|a-b|$. For $z \in \mathbb{R}^d$ and $r \geq 0$ the *open ball* with center z and radius r is $U_r(z) = \{x \in \mathbb{R}^d : \|x-z\| < r\}$ and the *closed ball* is $B_r(z) = \{x \in \mathbb{R}^d : \|x-z\| \leq r\}$, which we alternatively write as $B(z,r)$ whenever we want to stress the dependence on r.

We refer to the elements of $\mathcal{X}$ as *points*, to a finite subset $\mathcal{D} \subseteq \mathcal{X}$ as a *data set*, and to the elements of $\mathcal{D}$ as *objects* or *data points*, or also simply points, or *sample points* if $\mathcal{D}$ is obtained by sampling from some probability space.

1.3 Ordinal data and its different types

In a setting of ordinal distance information we do not have access to ι for evaluating dissimilarities between arbitrary objects directly. Instead, we only get to see a collection of binary answers to some dissimilarity comparisons

$$\iota(A,B) \overset{?}{<} \iota(C,D). \tag{1.1}$$

Note that we always assume the existence of a fixed dissimilarity function ι based on which (1.1) is evaluated. As in a "standard" cardinal setting, the success of all our attempts to do machine learning will crucially depend on the meaningfulness of ι, that is on whether ι captures the relevant properties of the data. However, the meaningfulness of ι is not what we are concerned with in this thesis.

Ordinal distance information appearing in practice is likely to be contaminated by noise, meaning that it comprises incorrect answers. This can result in *inconsistencies* in the ordinal data. For example, one could observe both an answer claiming that $\iota(A,B) < \iota(A,C)$ and a contradicting answer claiming that $\iota(A,B) > \iota(A,C)$, or the three inconsistent answers $\iota(A,B) < \iota(A,C)$, $\iota(A,C) < \iota(B,C)$, and $\iota(B,C) < \iota(A,B)$, where the contradiction is not immediately obvious. Many theoretical results (in particular regarding ordinal embedding—see the next section) are only established in the noise-free case, but any practical algorithm for ordinal data has to be be able to cope with noise.

While (1.1) is the most general form of a dissimilarity comparison, in practice one often deals with ordinal data comprising only answers to comparisons of a restricted form. For example, in scenarios of human-based computation and crowdsourcing it is particularly popular to show three objects A, B, and C at a time and to ask whether $\iota(A,B) < \iota(A,C)$ or $\iota(A,B) > \iota(A,C)$, that is, compared to (1.1), A equals D and serves as an *anchor point*. This gives rise to distinguishing between different types of ordinal data apart from the very general type comprising answers to arbitrary comparisons (1.1):

- Answers to dissimilarity comparisons

$$\iota(A,B) \overset{?}{<} \iota(A,C) \tag{1.2}$$

  are usually referred to as *similarity triplets*. Sometimes they are also referred to as *pairwise comparisons* (since B and C are compared regarding their distance from A), but we consider this term to be misleading due to its omnipresence in ranking. In the context of human-based computation and crowdsourcing, similarity triplets are usually preferred over ordinal data in its general form since it is assumed that answering (1.2) is even simpler than answering (1.1) for humans. For this reason, in the machine learning literature on ordinal distance information there is particular interest in this type of ordinal data (Jamieson and Nowak, 2011, Tamuz et al., 2011, van der Maaten and Weinberger, 2012, Wilber et al., 2014, Amid and Ukkonen, 2015, Heim et al., 2015, Amid et al., 2016, Jain et al., 2016, Haghiri et al., 2017). In this thesis, we are dealing with similarity triplets in Chapter 5.

- Another well-known and interesting type of ordinal distance information is the *directed, but unweighted k-nearest neighbor graph*, where $k \in \mathbb{N}$ is a parameter. Throughout this thesis we will use the notation "k-NN" as abbreviation for the term "k-nearest neighbor" whenever it is convenient. A directed, but unweighted k-NN graph on a data set $\mathcal{D}$ encodes knowledge about memberships to the sets of k nearest neighbors of data points: it has a directed, unweighted edge from an object A to an object B if and only if B is among the k nearest neighbors of A, that is B is among the k objects of $\mathcal{D} \setminus \{A\}$ for which the distance from A is smallest. This graph provides the ordinal dissimilarity relationships

$$\iota(V,N) < \iota(V,O)$$

  for objects V, N, and O such that N is adjacent to V in the graph, but O is not. In the machine learning literature on ordinal data the directed, but unweighted k-NN graph has been studied in Shaw and Jebara (2009), von Luxburg and Alamgir (2013), Terada and von Luxburg (2014), and, as a special case, in Hashimoto et al. (2015). In this thesis we will find it in Chapter 3. There we study the problem of estimating the intrinsic dimension of a data set from this type of ordinal distance information.

- Information about the outlier or about the most central object in a triple of objects: A further type of ordinal distance information are collections of statements

  $$\textit{Object } A \textit{ is the outlier within the triple of objects } (A,B,C), \tag{$\boxplus$}$$

  where (A,B,C) can be any triple of pairwise distinct objects in the data set $\mathcal{D}$ and object A being outlier within the triple of objects (A,B,C) means that

  $$\big(\iota(B,C) < \iota(A,B)\big) \;\;\wedge\;\; \big(\iota(B,C) < \iota(A,C)\big).$$

  Hence each statement of the kind ($\boxplus$) is equivalent to two similarity triplets. This type of ordinal data has been studied in Heikinheimo and Ukkonen (2013) and Ukkonen et al. (2015). In Chapter 4 we examine a closely related type, namely collections of statements of the form

  $$\textit{Object } A \textit{ is the most central object within the triple of objects } (A,B,C) \tag{$\star$}$$

  with the meaning that

  $$\big(\iota(A,B) < \iota(B,C)\big) \;\;\wedge\;\; \big(\iota(A,C) < \iota(B,C)\big).$$

  Statements of the kind ($\boxplus$) or ($\star$) can easily be collected via crowdsourcing. We will illustrate this by an example in Section 4.1.

So far we ignored the possibility of equal dissimilarity values $\iota(A,B) = \iota(C,D)$. We could consider variants of the listed types of ordinal data in which we allow for additional information on ties in dissimilarity comparisons. However, such a distinction is mainly relevant for theoretical considerations, in particular on ordinal embedding (see the next section). It hardly plays a role in practice: if we cannot access ι directly, it seems unlikely that we can observe precise equality of two dissimilarity values. Hence, at this point and in the practical parts of this thesis we omit the possibility of equal dissimilarity values.

The different types of ordinal data differ in their *information content*. We use this term informally and define it implicitly by referring to the following fact: given the (correct) answers to all possible dissimilarity comparisons of the form (1.1) for a data set $\mathcal{D}$ (there are $\mathcal{O}(|\mathcal{D}|^4)$ of them), we are also given all similarity triplets for $\mathcal{D}$ (there are $\mathcal{O}(|\mathcal{D}|^3)$ of them), from which we can readily build the k-NN graph on $\mathcal{D}$ or deduce all statements of the form ($\boxplus$) or ($\star$) (there are $\mathcal{O}(|\mathcal{D}|^3)$ different statements for each form). In general, the converses are not true, meaning that one cannot infer the answers to all dissimilarity comparisons (1.1) given all similarity triplets, for example. It is not clear how to quantify information content and how much is lost or gained in going from one type of ordinal data to another. Neither is it clear how the information content of ordinal data of a particular type reduces as the number of answered dissimilarity comparisons decreases. It seems plausible to expect that this highly depends on the structural properties of the dissimilarity function ι and the underlying set $\mathcal{X}$.

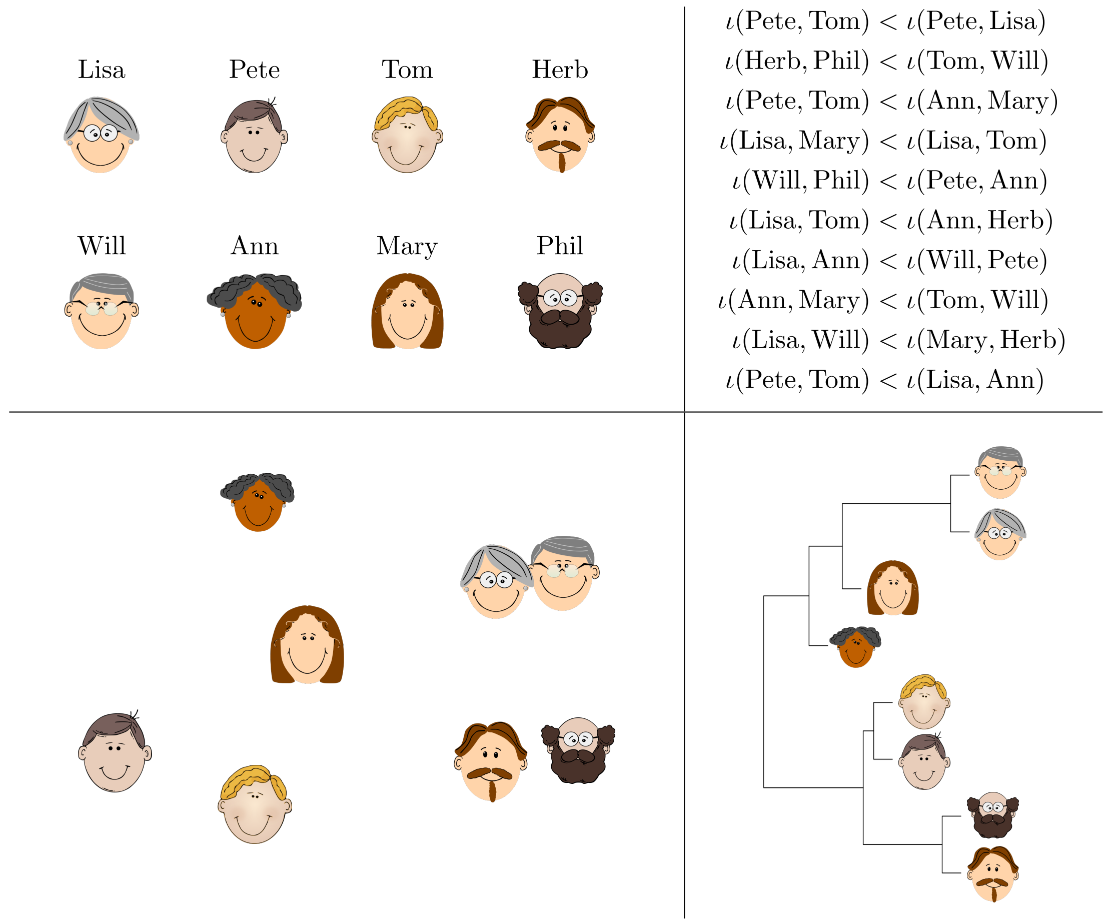
In any case, we have to hope that *abundant* ordinal data comprises redundant answers. By abundant ordinal data we mean ordinal data comprising the answers to all possible dissimilarity comparisons according to the type of ordinal data under consideration. Collecting and handling abundant ordinal data is prohibitive in practice due to the large number of possible dissimilarity comparisons. Ideally, only a small subset of the answers to all possible comparisons contains already the bulk of valuable information. This gives rise to distinguishing between a *batch setting* and an *active setting* in the study of algorithms for ordinal distance information: while in a batch setting we are given the ordinal data a priori, in an active setting we are allowed to query ordinal relationships, trying to do it in such a way as to exploit redundancy and to maximize information content per query (Jamieson and Nowak, 2011, Tamuz et al., 2011). Our algorithms that we present in Chapter 4 and Chapter 5 are designed for the general batch setting, but some of them could be combined with simple heuristics in order to obtain active versions (compare with Section 4.6).

1.4 Ordinal embedding

One general approach to deal with ordinal distance information is to construct an *ordinal embedding* of the data set $\mathcal{D}$, that is to map data points to points in a Euclidean space $\mathbb{R}^d$ such that the embedding (with respect to the Euclidean metric) preserves the given ordinal data "as well as possible". The dimension d of the space of the embedding is usually fixed a priori. We refer not only to the output of this procedure but also to the procedure itself as ordinal embedding. It is a way of transforming ordinal distance information back to a "standard" cardinal setting: once $\mathcal{D}$ is represented by points in $\mathbb{R}^d$, we can apply any machine learning algorithm designed for vector-valued data.

We illustrate the ordinal embedding approach with an example in Figure 1.2. In this example $\mathcal{D}$ consists of eight people, shown in the upper left part of the figure. We are provided with answers to ten dissimilarity comparisons of the general form (1.1) for these eight people, which are given in the upper right part. Two out of the ten comparisons are actually of the restricted form (1.2). Based on the provided answers we constructed an ordinal embedding of $\mathcal{D}$ in $\mathbb{R}^2$ and used it for clustering $\mathcal{D}$. The ordinal embedding is shown in the lower left part of the figure. It is a *perfect* ordinal embedding, that is it correctly reflects all the given ordinal constraints. The lower right part shows the clustering result in form of a dendrogram. It was obtained by applying single-linkage clustering (e.g., Manning et al., 2008, Chapter 17) based on Euclidean interpoint distances to the ordinal embedding and looks quite reasonable: Lisa and Will, both of them having grey hair and wearing glasses, are the first ones to be merged to one cluster, followed by Herb and Phil, who share an eye-catching beard.

Our definition of an ordinal embedding as a Euclidean point configuration that preserves the given ordinal data "as well as possible" is only informal. In the literature



$\iota(\text{Pete}, \text{Tom}) < \iota(\text{Pete}, \text{Lisa})$
$\iota(\text{Herb}, \text{Phil}) < \iota(\text{Tom}, \text{Will})$
$\iota(\text{Pete}, \text{Tom}) < \iota(\text{Ann}, \text{Mary})$
$\iota(\text{Lisa}, \text{Mary}) < \iota(\text{Lisa}, \text{Tom})$
$\iota(\text{Will}, \text{Phil}) < \iota(\text{Pete}, \text{Ann})$
$\iota(\text{Lisa}, \text{Tom}) < \iota(\text{Ann}, \text{Herb})$
$\iota(\text{Lisa}, \text{Ann}) < \iota(\text{Will}, \text{Pete})$
$\iota(\text{Ann}, \text{Mary}) < \iota(\text{Tom}, \text{Will})$
$\iota(\text{Lisa}, \text{Will}) < \iota(\text{Mary}, \text{Herb})$
$\iota(\text{Pete}, \text{Tom}) < \iota(\text{Lisa}, \text{Ann})$

Figure 1.2: An illustration of the ordinal embedding approach. Top left: The data set $\mathcal{D}$ consists of eight nice-looking people. Top right: Ordinal data for $\mathcal{D}$. Bottom left: An ordinal embedding of $\mathcal{D}$ in $\mathbb{R}^2$. Bottom right: Dendrogram obtained by applying single-linkage clustering to the ordinal embedding.[2]

there appear various more restricted or more formal definitions that are appropriate for the respective purpose (see below). Often, ordinal embedding is used as a synonym for *ordinal multidimensional scaling* (ordinal MDS; also known as *non-metric* MDS). Ordinal MDS has a long history in the psychometric community dating back to the 1960s (Shepard, 1962a,b, Kruskal, 1964a,b) and is a variant of the more famous *metric* multidimensional scaling (see the monograph Borg and Groenen, 2005, about both ordinal and metric MDS as well as further variants and Chapter 2 of Young, 1987, about the history of MDS). Generally speaking, given n objects $o_1, \ldots, o_n$ and dissimilarity values $\iota(o_i, o_j)$ for some pairs (o_i, o_j), multidimensional scaling aims at finding points $p_1, \ldots, p_n \in \mathbb{R}^d$ such that

$$f(\iota(o_i, o_j)) \approx \tilde{\iota}(p_i, p_j) \qquad (1.3)$$

[2] The images were found on openclipart.org and are in the public domain.

for all given dissimilarity values and the approximation is “optimal” regarding the choice of points $p_1, \ldots, p_n$ and of a function $f : [0, \infty) \to [0, \infty)$ out of a given function class. In principle, $\tilde{\iota}$ could be any dissimilarity function on $\mathbb{R}^d$, but most often only the Euclidean metric is considered and so do we in this thesis: from now on $\tilde{\iota}(p_i, p_j) = \|p_i - p_j\|$. In metric MDS, the function class from which f can be chosen is given by a parametric model, for example, all affine functions with non-negative slope $f(x) = a \cdot x + b$ governed by parameters $a, b \in \mathbb{R}$, $a \geq 0$, or consists of only a single function, for example, $f(x) = x$. In ordinal MDS, f is only restricted to be within the class of (either weakly or strictly) increasing functions. Optimality is measured by a so-called *stress function*. This could be the stress-1 function introduced by Kruskal (1964a) and given as

$$\sigma_1 = \sigma_1(p_1, \ldots, p_n, f) = \sqrt{\frac{\sum_{(i,j):\ \iota(o_i,o_j) \text{ is given}} [f(\iota(o_i, o_j)) - \|p_i - p_j\|]^2}{\sum_{(i,j):\ \iota(o_i,o_j) \text{ is given}} \|p_i - p_j\|^2}}.$$

In this expression the numerator of the fraction quantifies the deviations in the approximations (1.3). The denominator serves the purpose of normalization such that $\sigma_1(p_1, \ldots, p_n, f) = \sigma_1(c \cdot p_1, \ldots, c \cdot p_n, c \cdot f)$, $c \neq 0$, and prevents the degenerate point configuration $p_1 = p_2 = \ldots = p_n$. Other stress functions are designed on a similar basis. Solving the MDS problem now consists of minimizing the stress function under consideration.

The difference between ordinal embedding according to our informal definition and ordinal MDS is the presence of actual dissimilarity values $\iota(o_i, o_j)$ in the latter—although they influence the problem only regarding their ordinal relationships. Clearly, from the dissimilarity values $\iota(o_i, o_j)$ we can readily derive ordinal distance information, but given ordinal data we can only find dissimilarity values representing this ordinal data if the ordinal data is indeed consistent with a dissimilarity function (compare with Section 1.3). Even if we deal with consistent ordinal data, by representing this data by numerical dissimilarity values we might introduce new ordinal relationships that ordinal MDS takes into account. For example, $\iota(A, B) < \iota(A, C)$ and $\iota(E, F) < \iota(E, G)$ could be represented by setting $\iota(A, B) = 1$, $\iota(A, C) = 2$, $\iota(E, F) = 3$, $\iota(E, G) = 4$, but this suggests that $\iota(A, C) < \iota(E, F)$. Hence one may consider ordinal embedding a generalization of ordinal MDS, and for that reason Agarwal et al. (2007) have named their algorithm for ordinal embedding *generalized non-metric multidimensional scaling* (GNMDS). GNMDS has not been the first algorithm for ordinal embedding (Johnson, 1973, introduced a kind of stress function assuming the presence of dissimilarity values that can actually be fed with arbitrary ordinal data, whether consistent or inconsistent with a dissimilarity function), but it seems that Agarwal et al. (2007) have been the first to point out the subtly more general character of their algorithm. Essentially, GNMDS tries to find a point configuration such that every given ordinal relationship $\iota(o_i, o_j) < \iota(o_{i'}, o_{j'})$ is ideally reflected by $\|p_i - p_j\|^2 + 1 \leq \|p_{i'} - p_{j'}\|^2$ and the average distortion $1 + \|p_i - p_j\|^2 - \|p_{i'} - p_{j'}\|^2$ of not ideally reflected relationships is minimal. The purpose of requiring squared interpoint distances to differ by at least one for satisfactorily reflecting $\iota(o_i, o_j) < \iota(o_{i'}, o_{j'})$ is to prevent the degenerate embedding $p_1 = p_2 = \ldots = p_n$.

In recent years, a number of algorithms for ordinal embedding, all of them pretty similar in spirit to GNMDS, have been published in the machine learning community

(Shaw and Jebara, 2009, Tamuz et al., 2011, van der Maaten and Weinberger, 2012, Terada and von Luxburg, 2014, Amid and Ukkonen, 2015, Heim et al., 2015, Amid et al., 2016, Jain et al., 2016). The method by Shaw and Jebara (2009) only works for ordinal data in the form of a directed, but unweighted k-nearest neighbor graph. The algorithms by Tamuz et al. (2011), van der Maaten and Weinberger (2012), Amid and Ukkonen (2015), Amid et al. (2016), and Jain et al. (2016) are designed for similarity triplets (compare with Section 1.3). The method by Amid and Ukkonen (2015) actually produces a number of ordinal embeddings at the same time, each corresponding to a different dissimilarity function ι based on which a distance comparison (1.2) might have been evaluated. This corresponds to a generalization of the setup that we consider in this thesis. In Heim et al. (2015), GNMDS and one of the algorithms by van der Maaten and Weinberger (2012) are adapted from the batch setting to an online setting, in which similarity triplets are observed in a sequential way (other than in an active setting without the possibility of choosing which dissimilarity comparisons are evaluated). Although offering a seemingly appealing way to deal with machine learning problems in a setting of ordinal distance information, we argue in Chapter 4 that all these algorithms come with a number of shortcomings (e.g., a very high running time) and that there is a need for algorithms that try to solve machine learning problems based on ordinal data directly, without constructing an ordinal embedding as an intermediate step.

There has also been made significant progress in the theory of ordinal embedding in recent years. In our way of speaking, some of the results are actually rather on ordinal MDS since they assume noise-free ordinal data comprising answers to all possible dissimilarity comparisons (1.1), which allows to specify the original dissimilarity values $\iota(o_i, o_j)$ up to a monotone transformation.

Bilu and Linial (2005) consider data sets comprising n objects such that $\iota(o_i, o_j) = 0$ if and only if $i = j$ and all other pairwise dissimilarity values are distinct, that is

$$\iota(o_i, o_j) \neq \iota(o_{i'}, o_{j'}), \quad i < j, i' < j', (i \neq i') \vee (j \neq j'). \tag{1.4}$$

They define an ordinal embedding $p_1, \dots, p_n$ of such a data set by requiring

$$\forall i \leq j, i' \leq j' : \quad \iota(o_i, o_j) < \iota(o_{i'}, o_{j'}) \quad \Leftrightarrow \quad \|p_i - p_j\| < \|p_{i'} - p_{j'}\|. \tag{1.5}$$

Bilu and Linial show that for any data set there exists an ordinal embedding according to this definition if the dimension d of the space of the embedding is just large enough. However, it does not have to be larger than the size of the data set, that is there is always a feasible dimension $d \leq n - 1$. They claim that this *existence result* has already been folklore. Actually, it also holds if we drop the assumption (1.4). An ordinal embedding according to (1.5) then necessarily satisfies

$$\forall i \leq j, i' \leq j' : \quad \iota(o_i, o_j) = \iota(o_{i'}, o_{j'}) \quad \Leftrightarrow \quad \|p_i - p_j\| = \|p_{i'} - p_{j'}\|.$$

Bilu and Linial also prove that almost every data set (in a certain-well defined sense) requires the embedding dimension d to be almost as large as n, that is $d \in \Omega(n)$.

A main contribution of this thesis is to establish the first *uniqueness result* for ordinal embedding. Its formulation is rather involved and will be presented in detail in

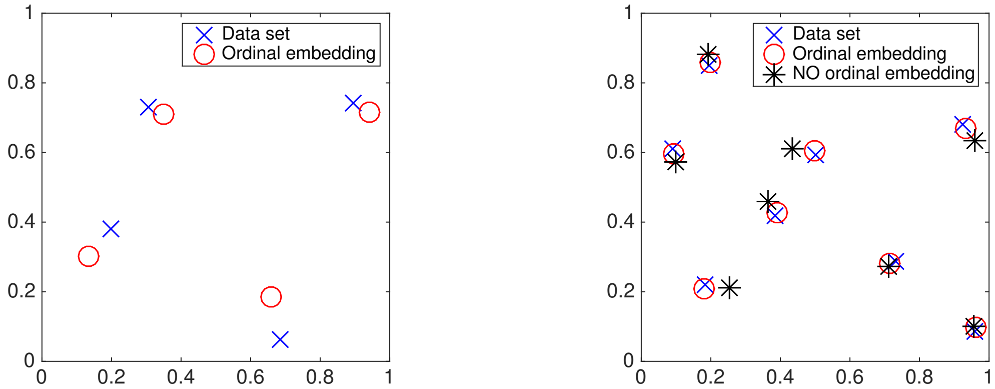


Figure 1.3: An illustration of Shepard's conjecture, which claims that the possible amount of deforming an ordinal embedding decreases as the number of objects in the data set increases. See the running text for an explanation.

Chapter 2. Here we just want to sketch its general idea. We define an ordinal embedding similarly to Bilu and Linial (2005) by requiring

$$\forall i \leq j, i' \leq j' : \quad \iota(o_i, o_j) < \iota(o_{i'}, o_{j'}) \quad \Rightarrow \quad \|p_i - p_j\| < \|p_{i'} - p_{j'}\|. \tag{1.6}$$

Observe that such an ordinal embedding is never uniquely determined: Given one ordinal embedding, we can simply construct another one by mapping every point p_i to $\tilde{p}_i = f(p_i)$, where f is an arbitrary similarity transformation on $\mathbb{R}^d$ (a similarity transformation is some composition of rescaling, translation, rotation, and reflection and preserves all pairwise distances up to a constant multiple). Also, we could simply shift every point of the given ordinal embedding to a slightly different location $\tilde{p}_i = p_i + \varepsilon_i$ ($\varepsilon_i \in \mathbb{R}^d$ with $\|\varepsilon_i\|$ sufficiently small) and $\tilde{p}_1, \ldots, \tilde{p}_n$ would still satisfy (1.6). However, Shepard (1962b, 1966) observed that these two kinds of transformations might be the only reasons for ambiguity. Even more, he claimed and experimentally confirmed that for the second kind the largest possible value of $\max_{i=1,\ldots,n} \|\varepsilon_i\|$ gets smaller the larger n gets. We illustrate this by an example in Figure 1.3: It shows two data sets consisting of four points (left plot) and eight points (right plot), respectively, as well as further point configurations along with each data set. In the left plot, we can see an ordinal embedding of the data set satisfying (1.6) (with ι equaling $\|\cdot\|_{\mathbb{R}^2}$). The ordinal embedding resembles the data set, but still there is some noticeable difference. In the right plot, we can see an ordinal embedding that almost perfectly coincides with the data set and another point configuration that is only slightly different (and more similar to the data set than the ordinal embedding to the corresponding data set in the left plot). However, this point configuration is not an ordinal embedding of the data set since it does not reflect some of the data set's ordinal relationships, that is (1.6) does not hold. In Chapter 2 we provide a mathematical formalization of Shepard's claim and prove that, under some assumptions, it is indeed true. These results have been published in Kleindessner and von Luxburg (2014).

Ordinal embeddings defined as in (1.5) or (1.6) are required to preserve all ordinal relationships of the general form (1.1). It is natural and also of practical interest to address existence and uniqueness of ordinal embeddings that are required to preserve only some ordinal relationships or ordinal data of a different type (compare with Section 1.3). For example, it is known that deciding whether there exists an ordinal embedding of a data set $\mathcal{D}$ into the real line $\mathbb{R}$ that preserves all statements of a given collection of statements ($\star$) about $\mathcal{D}$ (allowed to be noisy) is NP-complete (Opatrný, 1979). Regarding uniqueness, we actually show in Chapter 2 that Shepard's conjecture also holds true if we relax the requirement of an ordinal embedding to preserve all ordinal relationships in a certain way. Our results have further been generalized by Arias-Castro (2015). We will summarize his generalizations in detail in Section 2.5 of Chapter 2, but essentially he shows asymptotic uniqueness of ordinal embedding based on similarity triplets, the directed, but unweighted k-NN graph with some additional ordinal relationships, and in a so-called landmark design. The asymptotic uniqueness of ordinal embedding based on the directed, but unweighted k-NN graph has also been outlined in Terada and von Luxburg (2014). In Chapter 4 we conjecture the uniqueness property to hold for ordinal embedding based on statements of the kind ($\star$) too, as long as the intrinsic dimension of the data set is greater than one.

Jamieson and Nowak (2011) consider ordinal embeddings that are required to preserve all similarity triplets for a data set. Assuming the existence of an ordinal embedding in $\mathbb{R}^d$, they study the question of how large a subset of all similarity triplets has to be such that an embedding preserving the triplets in the subset automatically preserves all similarity triplets. Jamieson and Nowak show that the subset has to contain at least $\Omega(dn \log n)$ many similarity triplets. If the subset can be chosen adaptively, this lower bound is conjectured to be tight, whereas $\Omega(n^3)$ many similarity triplets are needed if the subset is chosen uniformly at random and it should be proper with probability greater than 1/2. The work of Jamieson and Nowak (2011) is concerned with ordinal embeddings exactly preserving all similarity triplets, other than the one by Jain et al. (2016). In the framework of empirical risk minimization, Jain et al. consider the problem of learning an embedding that can be used to predict the answers to dissimilarity comparisons (1.2). Under a certain noise model, they show that $\omega(dn \log n)$ many similarity triplets, chosen uniformly at random, are sufficient for learning an embedding that has almost the same risk of predicting an incorrect answer as the original data set with high probability.

1.5 Overview of the results

Now we want to give a short summary of the results covered in the following chapters:

- **Chapter 2:** This chapter is purely theoretical. We formalize Shepard's claim about the uniqueness of ordinal embeddings (see Section 1.4) using the notion of *isotonic functions*. These functions transform one point configuration into another one such that all ordinal relationships of the general type (1.1) are preserved (we also consider some variants). Shepard's claim can be reformulated as stating that any isotonic function can be approximated by a similarity transformation and that the quality of the approximation is better the larger the number of points. This involves comparing approximation qualities for functions defined on different sets of points, and we deal with it as follows: We consider a sequence of points $x_1, x_2, x_3, \ldots \in \mathbb{R}^d$ and a sequence of isotonic functions $\varphi_1, \varphi_2, \varphi_3, \ldots$ such that φ_n is defined on $x_1, x_2, \ldots, x_n$. Under some assumptions on the sequence of points and assuming that all functions φ_n map to the same bounded ball $U_r(0) \subseteq \mathbb{R}^d$, we can prove that there exists a sequence of similarity transformations $S_1, S_2, S_3, \ldots$ mapping $\mathbb{R}^d$ to $\mathbb{R}^d$ such that

$$\max_{i=1,\ldots,n} \|\varphi_n(x_i) - S_n(x_i)\| \to 0 \quad \text{as} \quad n \to \infty.$$

We also prove that an isotonic function defined on infinitely many points (satisfying some assumptions) actually is a similarity transformation. Both results have been published in Kleindessner and von Luxburg (2014).

Our results not only prove Shepard's long-standing claim, but also yield an important and positive insight for doing machine learning in a setting of ordinal distance information. We have shown that at least for a *Euclidean data set* (meaning the data set consists of points in $\mathbb{R}^d$ and ι equals the Euclidean metric) with known intrinsic dimension d abundant ordinal data of the type (1.1) asymptotically contains all cardinal distance information up to rescaling. This gives hope that for such a data set we might not loose much information in going from cardinal to ordinal distance information. We may hope that, in principle, we are able to solve any machine learning problem that we can solve in a "standard" cardinal setting also when given only ordinal data. It is not clear though how to recover the interpoint distances algorithmically. Neither is it clear whether machine learning problems in a setting of ordinal distance information come with a significant increase in computational complexity compared to a cardinal setting. Our results can also serve as theoretical justification for the ordinal embedding approach to machine learning problems based on ordinal data, which consists of constructing an ordinal embedding and solving the problem on the embedding. They guarantee that two perfect ordinal embeddings based on abundant ordinal data of the type (1.1) cannot be significantly different. Hence, for data sets that permit such perfect ordinal embeddings the final output of the embedding approach should not depend on the ordinal embedding at hand.

- **Chapter 3:** In the third chapter we deal with estimating the intrinsic dimension of a Euclidean data set when only given its directed, but unweighted k-nearest neighbor graph. This chapter is based on Kleindessner and von Luxburg (2015). We provide two estimators: a naive one and a more elaborate one that clearly outperforms the former. We can prove both estimators to be statistically consistent, and so this chapter is interesting from both a theoretical and a practical point of view. The consistency result is particularly appealing when combining it with the result of Chapter 2: we do no longer need to assume the intrinsic dimension of a Euclidean data set to be known when claiming that "we might not loose much information in going from cardinal to ordinal distance information".

- **Chapter 4:** This chapter is rather practical. We present algorithms for the machine learning problems of medoid estimation, outlier identification, classification, and clustering when the only given information about a data set $\mathcal{D}$ is a collection of statements of the kind ($\star$) (compare with Section 1.3). Our algorithms are *direct methods*, that is they do not construct an ordinal embedding of $\mathcal{D}$ as an intermediate step, and hence avoid some of the drawbacks inherent to such an embedding approach. Our algorithms are simple, are much faster than an ordinal embedding approach, and can easily and highly efficiently be parallelized. They are based on the insight that statements of the kind ($\star$) are intimately related to two well-known tools from multivariate statistics and computer vision, respectively: the lens depth function and the k-relative neighborhood graph. In a number of experiments we demonstrate the usefulness of our proposed methods. The contributions of Chapter 4 can also be found in Kleindessner and von Luxburg (2017).

- **Chapter 5:** The last chapter is practical too. We present two data-dependent kernel functions defined on a data set $\mathcal{D}$ that can be computed from an arbitrary collection of similarity triplets for the objects in $\mathcal{D}$. Unlike the algorithms proposed in Chapter 4, which are designed for specific tasks, these kernel functions provide a generic means for solving machine learning problems based on ordinal distance information since they can be used to apply any kernel method to $\mathcal{D}$. They avoid some of the drawbacks inherent to an embedding approach. In several experiments we demonstrate the meaningfulness of our kernel functions and study their performance when combined with a kernel method. This chapter is based on Kleindessner and von Luxburg (2016).

1.6 List of publications

As indicated in the previous section, this thesis is based on the following papers:

- M. Kleindessner and U. von Luxburg. Uniqueness of ordinal embedding. In M. F. Balcan, V. Feldman, and C. Szepesvári, editors, *Proceedings of The 27th Conference on Learning Theory (COLT)*, pages 40-67, 2014. FINALIST BEST STUDENT PAPER AWARD.

- M. Kleindessner and U. von Luxburg. Dimensionality estimation without distances. In G. Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 471-479, 2015.

- M. Kleindessner and U. von Luxburg. Lens depth function and k-relative neighborhood graph: versatile tools for ordinal data analysis. *Journal of Machine Learning Research*, 18(58):1-52, 2017.

- M. Kleindessner and U. von Luxburg. Kernel functions based on triplet similarity comparisons. Preprint to be found on arXiv (1607.08456 [stat.ML]), 2016.

Chapter 2

Asymptotic uniqueness of ordinal embeddings

Given all ordinal relationships of the type (1.1) for some data set $\mathcal{D}$ comprising n objects, Shepard (1962b, 1966) claimed and experimentally confirmed that "as n increases, the ordinal data is found to determine a spatial representation of the objects more and more nearly to within a general similarity transformation" (compare with Section 1.4; the quote is taken from the abstract of Shepard, 1966). In his seminal papers, Shepard also provided considerations about infinite point sets filling up convex regions of $\mathbb{R}^d$ as supporting evidence, and further simulation experiments have confirmed his claim as well (Young, 1970, Sherman, 1972). It seems that Shepard's claim has become folklore then (see Section 2.2 of Borg and Groenen, 2005, as well as Section 4.13.2 of Dattorro, 2005, and page 3 of Jamieson and Nowak, 2011). However, it had never been proved, neither had it been properly formalized.

In this chapter we provide a formalization of Shepard's claim assuming $\mathcal{D}$ to be Euclidean. We subsequently prove the claim to indeed be true. We also pick up on Shepard's considerations of infinite data sets and end up with a more general result compared to what is known from the literature. All our proofs are elementary in the sense that we do not apply any heavy mathematical machinery. However, details are delicate and require a careful treatment. Inspired by our work, Arias-Castro (2015) has generalized our results in various ways. We will summarize his results in Section 2.5.

2.1 Setup, definitions, and notation for Chapter 2

We start with the definition of the two central notions in this chapter, ordinal embeddings and isotonic functions. We will see below that these two notions are intimately related. Note that we define ordinal embeddings only for Euclidean data sets.

Definition 2.1 (Ordinal embedding). *Consider two data sets $\mathcal{Y}_n = \{y_1, \ldots, y_n\} \subseteq \mathbb{R}^d$ and $\mathcal{Z}_n = \{z_1, \ldots, z_n\} \subseteq \mathbb{R}^d$. $\mathcal{Z}_n$ is an* ordinal embedding *of $\mathcal{Y}_n$ if for all $1 \leq i, j, k, l \leq n$,*

$$\|y_i - y_j\| < \|y_k - y_l\| \Rightarrow \|z_i - z_j\| < \|z_k - z_l\|. \tag{2.1}$$

$\mathcal{Z}_n$ is a weak ordinal embedding *of $\mathcal{Y}_n$ if (2.1) holds for all $1 \leq i, j, k, l \leq n$ with $i = k$. $\mathcal{Z}_n$ is a* strong ordinal embedding *of $\mathcal{Y}_n$ if (2.1) holds for all $1 \leq i, j, k, l \leq n$ and additionally $\|y_i - y_j\| = \|y_k - y_l\| \Rightarrow \|z_i - z_j\| = \|z_k - z_l\|$ for all $1 \leq i, j, k, l \leq n$.*

Definition 2.2 (Isotonic functions)**.** *Let $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ and $f : \Omega \to \mathbb{R}^d$ be an arbitrary function. f is a* similarity *if there is $\lambda > 0$ such that for all $x, y \in \Omega$ we have $\|f(x) - f(y)\| = \lambda \|x - y\|$. f is* isotonic *or an* isotony *if for all $x, y, z, w \in \Omega$,*

$$\|x - y\| < \|z - w\| \Rightarrow \|f(x) - f(y)\| < \|f(z) - f(w)\|.$$

f is weakly isotonic *if this property only holds for $x, y, z, w \in \Omega$ with $x = z$. f is* strongly isotonic *if it is isotonic and additionally satisfies $\|x - y\| = \|z - w\| \Rightarrow \|f(x) - f(y)\| = \|f(z) - f(w)\|$ for all $x, y, z, w \in \Omega$.*

We say that f is locally *a similarity / (weakly / strongly) isotonic if for each point $x \in \Omega$ there exists a neighborhood $U(x)$ in Ω such that $f|_{U(x)}$ has the corresponding property. If we want to emphasize that a function $f : \Omega \to \mathbb{R}^d$ has a property not only locally but on all of Ω, we sometimes say that f is* globally *a similarity / (weakly / strongly) isotonic.*

Let us mention some obvious but important observations. Similarities $f : \mathbb{R}^d \to \mathbb{R}^d$ are nothing else than the well-known similarity transformations given by $f(x) = \lambda O x + b$ for some orthogonal matrix $O \in \mathbb{R}^{d \times d}$ and an offset $b \in \mathbb{R}^d$. For general Ω, they are simply given by the restrictions of similarity transformations to Ω (see Lemma 2.14 in Section 2.6.1). We have

$$\text{similarity} \Rightarrow \text{strongly isotonic} \Rightarrow \text{isotonic} \Rightarrow \text{weakly isotonic},$$

but for general Ω none of the converses are true. Any weakly isotonic function is injective. If f is a similarity or a strong isotony, so is $f^{-1} : f(\Omega) \to \mathbb{R}^d$, but this does not necessarily hold for isotonies. A composition of similarities / (weak / strong) isotonies is again a similarity / (weak / strong) isotony.

Obviously, $z_1, \ldots, z_n$ is a (weak / strong) ordinal embedding of $y_1, \ldots, y_n$ if and only if the mapping $f : \{y_1, \ldots, y_n\} \to \{z_1, \ldots, z_n\}$ given by $f(y_i) = z_i$, $i = 1, \ldots, n$, is (weakly / strongly) isotonic. Shepard's claim can thus be reformulated as stating that any isotonic function between two finite point sets can be approximated by a similarity and that the approximation gets better the larger the number of points. In this chapter we deal with the general question of approximating (weakly / strongly) isotonic functions by similarities. Let us mention that it is well-known that any strongly isotonic function $f : \mathbb{R}^d \to \mathbb{R}^d$ defined on the full domain $\mathbb{R}^d$ actually is a similarity transformation. One can see this by exploiting properties of sphere-preserving mappings in Euclidean geometry (see McKemie and Väisälä, 1999, and also the argumentation in Shepard, 1966), by an elegant argument related to positive definite functions (Schoenberg, 1938), and also by the Beckman-Quarles theorem (Beckman and Quarles, Jr., 1953).

We conclude this section with introducing some notation for the rest of Chapter 2. For any non-empty subset $\emptyset \neq A \subseteq \mathbb{R}^d$ we denote its linear hull by $[A] = \{\sum_{i=1}^n \lambda_i a_i :$

$n \in \mathbb{N}, a_i \in A, \lambda_i \in \mathbb{R}\}$ and its affine hull by $\mathcal{H}(A) = \{\sum_{i=1}^n \lambda_i a_i : n \in \mathbb{N}, a_i \in A, \lambda_i \in \mathbb{R}, \sum_{i=1}^n \lambda_i = 1\}$. For $g = (g_1, \ldots, g_d), g' = (g'_1, \ldots, g'_d) \in \mathbb{R}^d$, by $g < g'$ we mean $g_i < g'_i$ for $i = 1, \ldots, d$. For a vector-valued function $f : X \to \mathbb{R}^d$ and $j = 1, \ldots, d$ we write f^j for the jth component of f. For two functions $f : X_1 \to \mathbb{R}^d$ and $g : X_2 \to \mathbb{R}^d$ and an arbitrary subset $X \subseteq X_1 \cap X_2$ we denote the supremum norm between f and g on X by $\|f - g\|_{\infty(X)} = \sup_{x \in X} \|f(x) - g(x)\|$. At some points we will speak of a cross-polytope. By this we mean the image $T(C)$ of the d-dimensional standard cross-polytope C, which is given by the convex hull of all permutations of $(\pm 1/0/0/\ldots/0) \in \mathbb{R}^d$, under some similarity transformation T.

2.2 Main results

In this section we present our main results. All proofs are deferred to the subsequent sections. Our key question is to what extent we can approximate isotonic functions by similarities. We distinguish the cases of isotonic functions defined on a finite set of points and isotonic functions defined on an infinite domain.

Our first result concerns the more interesting finite case. Essentially it is Shepard's claim: We consider $\mathcal{D}_n = \{x_1, \ldots, x_n\} \subseteq \mathbb{R}^d$ and an isotonic mapping $\varphi_n : \mathcal{D}_n \to \mathbb{R}^d$. Recall that $\varphi_n(\mathcal{D}_n) = \{\varphi_n(x_1), \ldots, \varphi_n(x_n)\}$ is an ordinal embedding of $\mathcal{D}_n$. Under some assumptions, we prove that φ_n can be approximated by a similarity transformation up to arbitrary precision as $n \to \infty$.

Theorem 2.3 (Isotonic on a finite set implies approximate similarity)**.**

1. *Global Version: Let $K = B_r(z) \subseteq \mathbb{R}^d$ be a closed and bounded ball (for some arbitrary $r > 0$, $z \in \mathbb{R}^d$). Let $(x_n)_{n \in \mathbb{N}}$ be a sequence of points $x_n \in K$ such that $\{x_n : n \in \mathbb{N}\}$ is dense in K. Let $0 < R < \infty$ and $(\varphi_n)_{n \in \mathbb{N}}$ be a sequence of isotonic functions $\varphi_n : \{x_1, \ldots, x_n\} \to U_R(0) \subseteq \mathbb{R}^d$. Then there exists a sequence $(S_n)_{n \in \mathbb{N}}$ of similarity transformations $S_n : \mathbb{R}^d \to \mathbb{R}^d$ such that*

$$\|S_n - \varphi_n\|_{\infty(\{x_1, \ldots, x_n\})} \to 0 \quad \textit{as} \quad n \to \infty. \tag{2.2}$$

2. *Local Version: More generally, let $K = \bigcup_{i=1}^k K_i \subseteq \mathbb{R}^d$ be a finite union of closed and bounded balls such that $\bigcup_{i=1}^k K_i^\circ$ is connected. Let $(x_n)_{n \in \mathbb{N}}$ be a sequence of points $x_n \in K$ such that $\{x_n : n \in \mathbb{N}\}$ is dense in K. Let $0 < R < \infty$ and $(\varphi_n)_{n \in \mathbb{N}}$ be a sequence of functions $\varphi_n : \{x_1, \ldots, x_n\} \to U_R(0) \subseteq \mathbb{R}^d$ such that*

$$\forall i \in \{1, \ldots, k\} : \varphi_n|_{\{x_1, \ldots, x_n\} \cap K_i} \; \textit{is isotonic.}$$

   Then there exists a sequence $(S_n)_{n \in \mathbb{N}}$ of similarity transformations $S_n : \mathbb{R}^d \to \mathbb{R}^d$ such that (2.2) holds.

Our proofs show that we can replace the set K in Part 1 of Theorem 2.3 by a cross-polytope or any closed and convex set that is a superset of a cross-polytope and a subset of the smallest ball containing the cross-polytope. Consequently, we can replace K in Part 2 by any finite union of such sets if we additionally assume that all these sets satisfy

$K_i \subseteq \overline{K_i^\circ}$. We will also see that in the one-dimensional case $d = 1$ the statements hold true if we only assume the functions φ_n to be weakly isotonic.

Note that the assumption that all functions φ_n map to the same bounded ball $U_R(0)$ is necessary. Otherwise the configurations of the image points could be blown up by a larger and larger constant such that the approximation error $\|S_n - \varphi_n\|_{\infty(\{x_1,\dots,x_n\})}$ was prevented from converging.

Our second result deals with the less surprising infinite case. It is known that any strongly isotonic function $f : \mathbb{R}^d \to \mathbb{R}^d$ actually is a similarity (see Section 2.1). We show that this holds true for only locally isotonic functions defined on more general infinite domains.

Theorem 2.4 (Isotonic on a dense set implies similarity)**.** *Let $\emptyset \neq G \subseteq \mathbb{R}^d$ be open and connected and $\Omega \subseteq G$ be a dense subset. Let $f : \Omega \to \mathbb{R}^d$ be a locally isotonic function. Then there exists a unique extension of f to a similarity transformation $F : \mathbb{R}^d \to \mathbb{R}^d$.*

2.3 Proof of Theorem 2.3 (the finite case)

2.3.1 Case $d = 1$

The case $d = 1$ is particularly simple. Our first lemma shows that any weakly isotonic function $f : \Omega \to \mathbb{R}$ (for arbitrary $\emptyset \neq \Omega \subseteq \mathbb{R}$) is either strictly increasing or decreasing.

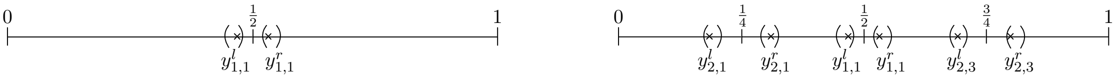
Lemma 2.5 (Weakly isotonic functions are monotonic)**.** *Let $\emptyset \neq \Omega \subseteq \mathbb{R}$ and $f : \Omega \to \mathbb{R}$ be a weakly isotonic function. Then f is either strictly increasing or strictly decreasing.*

Proof If f was neither strictly increasing nor decreasing, there would be $x < y < z \in \Omega$ such that either $(f(x) < f(y)) \wedge (f(y) > f(z))$ or $(f(x) > f(y)) \wedge (f(y) < f(z))$ would hold. However, for $x < y < z$ we have $|x - y| < |x - z|$ and $|z - y| < |z - x|$. Since f is weakly isotonic, it follows that $|f(x) - f(y)| < |f(x) - f(z)|$ and $|f(z) - f(y)| < |f(z) - f(x)|$ and hence either $f(x) < f(y) < f(z)$ or $f(z) < f(y) < f(x)$. ■

The following lemma is the main step of the proof of Theorem 2.3 in the one-dimensional case. It considers points that approximate a grid, and proves that this property remains intact after an isotonic mapping. See Figure 2.1 for an illustration.

Lemma 2.6 (Weakly isotonic maps approximately preserve a grid)**.** *Let $N \in \mathbb{N}$. For some $\varepsilon_1 < 1/2^{2N+1}$ set $\varepsilon_k = \varepsilon_1 2^{k-1}$, $2 \leq k \leq N$, and $\delta = \varepsilon_1/2$. For $k \in \{1, \dots, N\}$ and $i \in \{1, 3, \dots, 2^k - 1\}$ set $x_{k,i} = i/2^k$ and let $y^l_{k,i}$, $y^r_{k,i}$ be arbitrary elements of $(x_{k,i} - \varepsilon_k - \delta, x_{k,i} - \varepsilon_k)$ and $(x_{k,i} + \varepsilon_k, x_{k,i} + \varepsilon_k + \delta)$, respectively. Let $\varphi : \{0,1\} \cup \{y^m_{k,i} : m \in \{l,r\}, k \leq N, i \in \{1,3,\dots,2^k-1\}\} \to [0,1]$ be a weakly isotonic function with $\varphi(0) = 0$ and $\varphi(1) = 1$. Then it holds that*

$$\left|y^m_{k,i} - \varphi(y^m_{k,i})\right| < \frac{1}{2^N}, \quad m \in \{l,r\}, k \leq N, i \in \{1,3,\dots,2^k-1\}. \tag{2.3}$$



(a) $N = 1$ (b) $N = 2$

Figure 2.1: The idea of Lemma 2.6 is to place points in small intervals close to the grid points $i/2^N$ ($y^l_{k,i}$ on the left side, $y^r_{k,i}$ on the right side) in such a way that the ordinal constraints between all these points are sufficient to determine the grid cells they belong to, independent of their exact location within the intervals.

Proof (details can be found in Section 2.6.2) By induction over N we prove

$$\varphi(y^l_{k,i}) \in \left(\frac{2^{N-k}i - 1}{2^N}, \frac{2^{N-k}i}{2^N}\right), \quad \varphi(y^r_{k,i}) \in \left(\frac{2^{N-k}i}{2^N}, \frac{2^{N-k}i + 1}{2^N}\right),$$

for all $1 \leq k \leq N$, $i \in \{1, 3, \ldots, 2^k - 1\}$, which immediately implies (2.3). The basis is clear (see Figure 2.1(a)): Due to $\varphi(0) = 0$ and $\varphi(1) = 1$, φ is strictly increasing by Lemma 2.5 and hence $0 = \varphi(0) < \varphi(y^l_{1,1}) < \varphi(y^r_{1,1}) < \varphi(1) = 1$. Since $|y^l_{1,1} - 0| < |y^l_{1,1} - 1|$ and φ is weakly isotonic, we have $|\varphi(y^l_{1,1}) - \varphi(0)| < |\varphi(y^l_{1,1}) - \varphi(1)|$ and thus can conclude that $\varphi(y^l_{1,1}) \in (0, 1/2)$. In the same way we obtain $\varphi(y^r_{1,1}) \in (1/2, 1)$. We demonstrate the inductive step by proving that the statement also holds for $N = 2$ (see Figure 2.1(b)): We already know that $\varphi(y^l_{1,1}) \in (0, 1/2)$ and $\varphi(y^r_{1,1}) \in (1/2, 1)$. Furthermore, due to φ being strictly increasing, we have $0 < \varphi(y^l_{2,1}) < \varphi(y^r_{2,1}) < \varphi(y^l_{1,1}) < \varphi(y^r_{1,1}) < \varphi(y^l_{2,3}) < \varphi(y^r_{2,3}) < 1$. The choice of $(\varepsilon_k)_{1 \leq k \leq N}$ and δ guarantees that $|y^l_{2,1} - 0| < |y^l_{2,1} - y^l_{1,1}|$ and $|y^r_{2,1} - y^r_{1,1}| < |y^r_{2,1} - 0|$ leading to $|\varphi(y^l_{2,1}) - 0| < |\varphi(y^l_{2,1}) - \varphi(y^l_{1,1})|$ and $|\varphi(y^r_{2,1}) - \varphi(y^r_{1,1})| < |\varphi(y^r_{2,1}) - 0|$. This yields $2\varphi(y^l_{2,1}) < \varphi(y^l_{1,1}) < 1/2$ and $1/2 < \varphi(y^r_{1,1}) < 2\varphi(y^r_{2,1})$ and thus $\varphi(y^l_{2,1}) \in (0, 1/4)$ and $\varphi(y^r_{2,1}), \varphi(y^l_{1,1}) \in (1/4, 1/2)$. In the same way we can show that $\varphi(y^r_{2,3}) \in (3/4, 1)$ and $\varphi(y^l_{2,3}), \varphi(y^r_{1,1}) \in (1/2, 3/4)$. ■

Now it is straightforward to prove Theorem 2.3 for the case $d = 1$. Proposition 2.7 contains Part 1 of Theorem 2.3 and shows that in the one-dimensional case its assertion holds true if we only assume the functions φ_n to be weakly isotonic. The proof of Part 2 is the same as for the case $d \geq 2$, which follows later on.

Proposition 2.7 (Part 1 of Theorem 2.3 for $d = 1$)**.** *Let $I = [a, b]$ (for some $-\infty < a < b < \infty$) and let $(x_n)_{n\in\mathbb{N}}$ be a sequence of points $x_n \in I$ such that $\{x_n : n \in \mathbb{N}\}$ is dense in I. Let $0 < R < \infty$ and $(\varphi_n)_{n\in\mathbb{N}}$ be a sequence of weakly isotonic functions $\varphi_n : \{x_1, \ldots, x_n\} \to [-R, R]$. Then there exists a sequence $(S_n)_{n\in\mathbb{N}}$ of similarity transformations $S_n : \mathbb{R} \to \mathbb{R}$ with* (2.2).

Proof By appropriately rescaling the domain and the image of φ_n we may assume that $I = [0, 1]$ and that φ_n maps to $[0, 1]$ with $\varphi_n(0) = 0$, $\varphi_n(1) = 1$. We use Lemma 2.6 in order to show that φ_n for large values of n can be approximated by the identity, that is for all $\varepsilon > 0$ there exists $N_0 \in \mathbb{N}$ such that $\|\operatorname{id} - \varphi_n\|_{\infty(\{x_1,\ldots,x_n\})} < \varepsilon$ for all $n \geq N_0$. Choose $N \in \mathbb{N}$ such that $1/2^{N-2} < \varepsilon$. Since $\{x_n : n \in \mathbb{N}\}$ is dense in I, there exists $N_0 \in \mathbb{N}$ such that each of the intervals $(x_{k,i} - \varepsilon_k - \delta, x_{k,i} - \varepsilon_k)$ or $(x_{k,i} + \varepsilon_k, x_{k,i} + \varepsilon_k + \delta)$ as defined in Lemma 2.6 (for the chosen N) contains at least one element of $\{x_1, \dots, x_{N_0}\}$. If $n \geq N_0$, $y \in \{x_1, \dots, x_n\}$, and y is in one of the intervals, we immediately obtain $|y - \varphi_n(y)| < 1/2^N < \varepsilon$ according to (2.3). If y is not in one of the intervals, there exist two elements $\tilde{x}, \hat{x}$ of $\{x_1, \dots, x_{N_0}\}$ with $\tilde{x} < y < \hat{x}$ and $|\tilde{x} - \hat{x}| < 2/2^N$ that are in an interval. Using the monotonicity of φ_n we conclude that $|y - \varphi_n(y)| < 4/2^N < \varepsilon$. ■

2.3.2 Case $d \geq 2$

The case $d \geq 2$ is harder to deal with. Our basic idea is to show that an isotonic mapping $\varphi_n : \{x_1, \dots, x_n\} \to \mathbb{R}^d$, up to some rescaling, is an $\varepsilon(n)$-*nearisometry*, that is φ_n satisfies

$$\|x - y\| - \varepsilon(n) \leq \|\varphi_n(x) - \varphi_n(y)\| \leq \|x - y\| + \varepsilon(n), \quad x, y \in \{x_1, \dots, x_n\}. \tag{2.4}$$

Then, by a theorem of Alestalo et al. (2001), φ_n can be approximated by an isometry up to an error depending essentially only on $\varepsilon(n)$ and going to zero as $\varepsilon(n) \to 0$.

For proving that φ_n is an $\varepsilon(n)$-nearisometry we observe the following: since φ_n is isotonic, it is sufficient to prove (2.4) only for some pairs x, y such that $\|x - y\|$ is roughly uniformly distributed in $[0, \operatorname{diam}\{x_1, \dots, x_n\}]$. Hence, we would like to consider points close to a straight line segment with length equaling $\operatorname{diam}\{x_1, \dots, x_n\}$ and argue in a way similar to Lemma 2.6 that their relative positions along the line segment are almost preserved by an isotonic mapping. The problem is that, in general, there is no guarantee that the points are still close to a straight line segment after applying an isotony. However, assuming that there are points located close to the vertices of a cross-polytope and that these are "fixed" points (this is Assumption ($\sharp$) in the following lemma), we can show that this is the case and Lemma 2.6 can be generalized in the following sense. For the sake of understanding and simple presentation, here we just provide a compressed version of the lemma (see also Figure 2.2 for an explanation). A detailed version can be found in Section 2.6.3.

Lemma 2.8 (Under Assumption ($\sharp$) isotonic maps preserve an approximately straight line segment)**.** *Let $d \geq 2$. Let $N \in \mathbb{N}$ such that*

$$\omega = 24 \left(\frac{\Gamma(\frac{d}{2} + 1)}{\pi^{\frac{d}{2}}} \right)^{\frac{1}{d}} \left(\frac{1}{2^N - 1} \right)^{\frac{1}{d}} < \frac{1}{2(d-1)}$$

be fixed. Let U_s^+, U_s^-, $\widetilde{U}_s^+$, $\widetilde{U}_s^-$, $s = 1, \dots, d$, and $U_{k,i}^j$, $U_{k,i}^l$, $U_{k,i}^r$, $j \in \{2, \dots, d\}, 1 \leq k \leq N, i \in \{1, 3, \dots, 2^k - 1\}$, be open balls with some certain properties (see Section 2.6.3 for details). Let $X_s^+, X_s^- \in \mathbb{R}^d$, $s = 1, \dots, d$, be arbitrary elements of U_s^+ and U_s^-, respectively, $z_{k,i}^j \in \mathbb{R}^d$ be an arbitrary element of $U_{k,i}^j$, and $y_{k,i}^l$, $y_{k,i}^r \in \mathbb{R}^d$ be arbitrary elements of $U_{k,i}^l$ and $U_{k,i}^r$, respectively. Let $\varphi : \{X_1^+, X_1^-, \dots, X_d^+, X_d^-\} \cup \{z_{k,i}^j : k \leq N, i \in \{1, 3, \dots, 2^k - 1\}, j \in \{2, \dots, d\}\} \cup \{y_{k,i}^m : m \in \{l, r\}, k \leq N, i \in \{1, 3, \dots, 2^k - 1\}\} \to \mathbb{R}^d$ be an isotonic function and assume that

$$\varphi(X_s^+) \in \widetilde{U}_s^+, \quad \varphi(X_s^-) \in \widetilde{U}_s^-, \quad s = 1, \dots, d. \tag{$\sharp$}$$

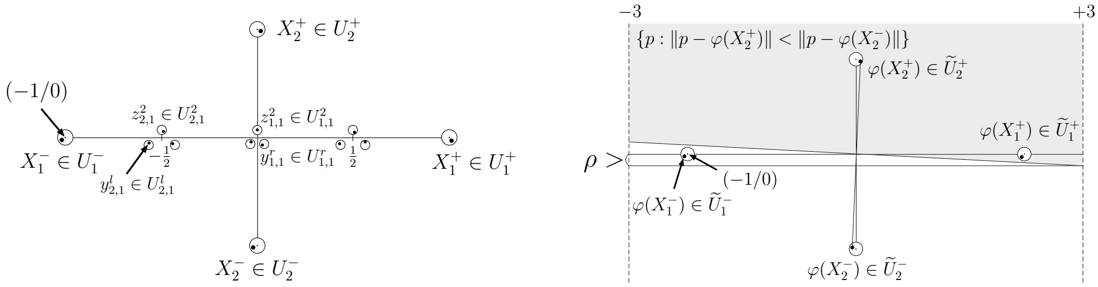

$X_2^+ \in U_2^+$
$(-1/0)$
$z_{2,1}^2 \in U_{2,1}^2$
$z_{1,1}^2 \in U_{1,1}^2$
$X_1^- \in U_1^-$
$-\frac{1}{2}$
$y_{1,1}^r \in U_{1,1}^r$
$\frac{1}{2}$
$X_1^+ \in U_1^+$
$y_{2,1}^l \in U_{2,1}^l$
$X_2^- \in U_2^-$
-3
$+3$
$\{p : \|p - \varphi(X_2^+)\| < \|p - \varphi(X_2^-)\|\}$
$\varphi(X_2^+) \in \widetilde{U}_2^+$
$\varphi(X_1^+) \in \widetilde{U}_1^+$
$\rho >$
$(-1/0)$
$\varphi(X_1^-) \in \widetilde{U}_1^-$
$\varphi(X_2^-) \in \widetilde{U}_2^-$

Figure 2.2: Explanation of Lemma 2.8 for $d = 2$. We consider an isotonic mapping φ defined on the following point set (left sketch): (i) $X_1^+, X_1^-, X_2^+, X_2^-$ are located in small balls around the vertices of a cross-polytope and assumed to be "fixed" under φ (this is Assumption ($\sharp$)). (ii) The points $y_{k,i}^l, y_{k,i}^r$ approximate a grid on the line segment between X_1^- and X_1^+ similarly as in Lemma 2.6 and are closer to X_2^- than to X_2^+. (iii) The points $z_{k,i}^2$ are close to the points $y_{k,i}^l$ and $y_{k,i}^r$, but are closer to X_2^+ than to X_2^-. Since φ is isotonic, the points $\varphi(y_{k,i}^l), \varphi(y_{k,i}^r)$ are closer to $\varphi(X_2^-)$ than to $\varphi(X_2^+)$ and hence $\varphi^2(y_{k,i}^l), \varphi^2(y_{k,i}^r) < \rho$, whereas for the points $\varphi(z_{k,i}^2)$ it is the other way round such that $\varphi^2(z_{k,i}^2) > -\rho$ (right sketch). However, $y_{k,i}^m$ ($m \in \{l, r\}$) and $z_{k,i}^2$ are close to each other and so are $\varphi(y_{k,i}^m)$ and $\varphi(z_{k,i}^2)$. We can conclude that all points $\varphi(y_{k,i}^m)$ are close to the first coordinate axis. This allows us to estimate the location of $\varphi(y_{k,i}^m)$ along similar lines as in Lemma 2.6.

Set $\gamma(-1) = \gamma(1) = \tilde{\alpha}_1$ and $\gamma(0) = \tilde{\alpha}_1 + \frac{d-1}{2}(\omega + \rho)$ (where $\tilde{\alpha}_1$ is the radius of the balls $\widetilde{U}_1^+, \widetilde{U}_1^-$ and ρ a small number depending on size and location of the balls $\widetilde{U}_s^+, \widetilde{U}_s^-$, $s = 2, \ldots, d$), and define for $2 \leq k \leq N$ and $i \in \{1, 3, \ldots, 2^k - 1\}$ the positive expression $\gamma(-1 + i/2^{k-1})$ recursively by

$$\gamma\left(-1 + \frac{i}{2^{k-1}}\right) = \frac{1}{2}\left(\gamma\left(-1 + \frac{i-1}{2^{k-1}}\right) + \gamma\left(-1 + \frac{i+1}{2^{k-1}}\right) + (d-1)(\omega + 2\rho)\right).$$

Let $N^* < N$ such that $N^* \cdot 2^{N^*} < \frac{1}{5(d+1)(\omega+\rho+\tilde{\alpha}_1)}$. Then we have

$$\left\| y_{k,i}^m - \varphi\left(y_{k,i}^m\right) \right\| < \gamma(x_{k,i}) + \omega + (d-1)(\omega + \rho) < 3d\sqrt{\omega}, \quad m \in \{l, r\}, \tag{2.5}$$

where $x_{k,i} = -1 + \frac{i}{2^{k-1}}$, for all $1 \leq k \leq N^*$ and $i \in \{1, 3, \ldots, 2^k - 1\}$.

Proof We prove that for all $1 \leq k \leq N^*$ and $i \in \{1, 3, \ldots, 2^k - 1\}$

$$\varphi(y_{k,i}^l) \in (x_{k,i} - \gamma(x_{k,i}) - \omega, x_{k,i} + \gamma(x_{k,i})) \times (-\rho - \omega, \rho)^{d-1},$$
$$\varphi(y_{k,i}^r) \in (x_{k,i} - \gamma(x_{k,i}), x_{k,i} + \gamma(x_{k,i}) + \omega) \times (-\rho - \omega, \rho)^{d-1}.$$

It is elementary to show that $\gamma(x_{k,i}) < \frac{1}{2}(d-1)\sqrt{3\omega}$, $1 \leq k \leq N^*$, and because of $y_{k,i}^l \in (x_{k,i} - \omega, x_{k,i}) \times (-\omega, 0)^{d-1}, y_{k,i}^r \in (x_{k,i}, x_{k,i} + \omega) \times (-\omega, 0)^{d-1}$ this immediately yields (2.5).

All points $y_{k,i}^l, y_{k,i}^r, z_{k,i}^j$ lie in the convex hull of the points $X_1^+, X_1^-, \ldots, X_d^+, X_d^-$. Since φ is isotonic and satisfies Assumption ($\sharp$), it is guaranteed that

$$\varphi(y_{k,i}^l), \varphi(y_{k,i}^r), \varphi(z_{k,i}^j) \in [-3, 3]^d. \tag{2.6}$$

We can prove $\varphi^j(y^l_{k,i}), \varphi^j(y^r_{k,i}) \in (-\rho-\omega, \rho)$, $j \in \{2, \dots, d\}$, as follows: Let j be fixed. For $m \in \{l, r\}$, $k \in \{1, \dots, N\}$, $i \in \{1, 3, \dots, 2^k - 1\}$ we have $\|y^m_{k,i} - X^-_j\| < \|y^m_{k,i} - X^+_j\|$ and $\|z^j_{k,i} - X^+_j\| < \|z^j_{k,i} - X^-_j\|$. Since φ is isotonic, it follows that $\|\varphi(y^m_{k,i}) - \varphi(X^-_j)\| < \|\varphi(y^m_{k,i}) - \varphi(X^+_j)\|$ and $\|\varphi(z^j_{k,i}) - \varphi(X^+_j)\| < \|\varphi(z^j_{k,i}) - \varphi(X^-_j)\|$. Making use of ($\sharp$) and (2.6) it is not difficult to show that $\varphi^j(y^m_{k,i}) < \rho$ and $\varphi^j(z^j_{k,i}) > -\rho$ (see the right side of Figure 2.2). The distance between any two points $z^j_{k_1,i_1}, z^j_{k_2,i_2}$ is larger than the distance between any two points $z^j_{k,i}, y^l_{k,i}$ (or $z^j_{k,i}, y^r_{k,i}$, respectively), that is for $m \in \{l, r\}$, $k \in \{1, \dots, N\}$, $i \in \{1, 3, \dots, 2^k - 1\}$ it holds that

$$\|z^j_{k,i} - y^m_{k,i}\| < \min\left\{\|u - v\| : u \neq v \in \left\{z^j_{\tilde{k},\tilde{i}} : \tilde{k} \leq N, \tilde{i} \in \left\{1, 3, \dots, 2^{\tilde{k}} - 1\right\}\right\}\right\}. \tag{2.7}$$

Let $m \in \{l, r\}$, $k_0 \leq N$, $i_0 \in \{1, 3, \dots, 2^{k_0} - 1\}$ be arbitrary and write $r = \|\varphi(z^j_{k_0,i_0}) - \varphi(y^m_{k_0,i_0})\|$. Due to (2.7) and φ being isotonic, all points $\varphi(z^j_{k,i})$ are located at distance larger than r to each other. It follows that the intersection of two balls (whether open or closed) with radius $r/2$ and centers $\varphi(z^j_{k_1,i_1})$ and $\varphi(z^j_{k_2,i_2})$, respectively, is empty. Recall (2.6). Due to ($\sharp$) and φ being isotonic we clearly have $r \leq 3$. Hence, with each point $\varphi(z^j_{k,i})$ at least a fraction of $1/2^d$ of the volume of the ball $U_{r/2}(\varphi(z^j_{k,i}))$ is contained in $[-3, 3]^d$ too. We can infer that

$$(2^N - 1)\frac{1}{2^d}\frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2} + 1)}\left(\frac{r}{2}\right)^d \leq 6^d,$$

or equivalently $r \leq \omega$. Hence, we have $|\varphi^j(z^j_{k_0,i_0}) - \varphi^j(y^m_{k_0,i_0})| \leq \|\varphi(z^j_{k_0,i_0}) - \varphi(y^m_{k_0,i_0})\| \leq \omega$ and finally obtain $\varphi^j(y^l_{k_0,i_0}), \varphi^j(y^r_{k_0,i_0}) \in (-\rho - \omega, \rho)$.

It remains to prove that for $1 \leq k \leq N^*$ and $i \in \{1, 3, \dots, 2^k - 1\}$,

$$\begin{aligned} \varphi^1(y^l_{k,i}) &\in (x_{k,i} - \gamma(x_{k,i}) - \omega, x_{k,i} + \gamma(x_{k,i}))\,, \\ \varphi^1(y^r_{k,i}) &\in (x_{k,i} - \gamma(x_{k,i}), x_{k,i} + \gamma(x_{k,i}) + \omega)\,. \end{aligned} \tag{2.8}$$

Similar to (2.7), we also have

$$\|y^l_{k,i} - y^r_{k,i}\| < \min\left\{\|u - v\| : u \neq v \in \left\{z^j_{\tilde{k},\tilde{i}} : \tilde{k} \leq N, \tilde{i} \in \left\{1, 3, \dots, 2^{\tilde{k}} - 1\right\}\right\}\right\},$$

and with the same argument as above we obtain $|\varphi^1(y^l_{k,i}) - \varphi^1(y^r_{k,i})| \leq \omega$. Now, (2.8) can be shown by induction over k similarly to the proof of Lemma 2.6. ■

The following lemma shows that the Assumption ($\sharp$), which says that points close to the vertices of a cross-polytope are mapped approximately to themselves, can be taken as satisfied if the isotonic function acts on sufficiently many points. See Figure 2.3 for an explanation. Again, here we just provide a compressed version of the lemma and the detailed version is in Section 2.6.3.

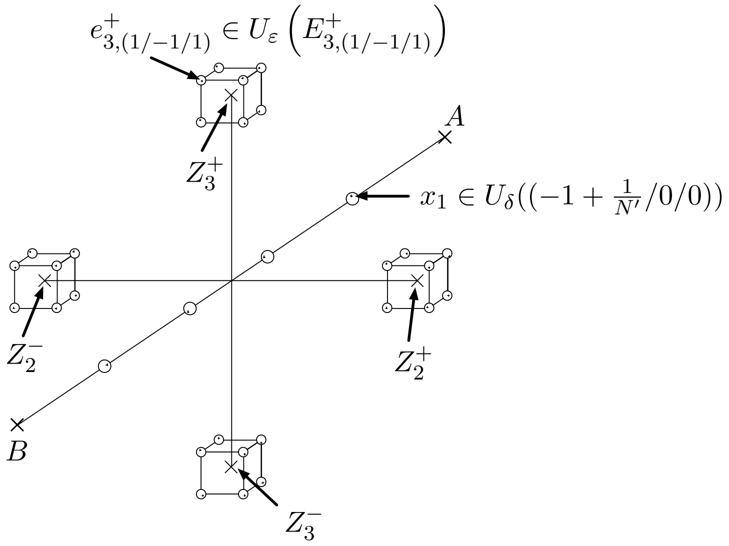

$e^+_{3,(1/-1/1)} \in U_\varepsilon\left(E^+_{3,(1/-1/1)}\right)$
A
Z_3^+
$x_1 \in U_\delta((-1+\frac{1}{N'}/0/0))$
Z_2^-
Z_2^+
B
Z_3^-

Figure 2.3: Explanation of Lemma 2.9 (the figure shows the setting of Lemma 2.9 for $d = 3$). We consider an isotonic map φ defined on the following point set: (i) A and B are opposite vertices of a cross-polytope. (ii) The points $e^-_{s,v}, e^+_{s,v}$ are located in small balls around the vertices of hypercubes placed around the remaining vertices of the cross-polytope (these remaining vertices are slightly shifted towards the center). (iii) Numerous points x_i are located in small balls which are placed equidistantly between A and B. This yields ordinal constraints that are sufficient to show that all points $e^-_{s,v}, e^+_{s,v}$ are "fixed" under φ up to some similarity transformation.

Lemma 2.9 (Assumption (♯) can be taken as satisfied)**.** *Let $d \geq 2$. Let $N' \in \mathbb{N}$ such that*

$$\omega' = 32\left(\frac{\Gamma(\frac{d}{2}+1)}{\pi^{\frac{d}{2}}}\right)^{\frac{1}{d}} \frac{1}{\sqrt[d]{N'}}$$

is sufficiently small and $r' < 1$ and $\eta, \delta, \varepsilon > 0$ be appropriately chosen real numbers (see Section 2.6.3 for details). Define points $A, B \in \mathbb{R}^d$ and $Z_s^-, Z_s^+ \in \mathbb{R}^d$, $s \in \{2, \dots, d\}$, by

$$A = (-1/0/\dots/0), \quad B = (1/0/\dots/0), \quad Z_2^- = \left(0/-r'/0/0/\dots\right),$$
$$Z_2^+ = \left(0/r'/0/0/\dots\right), \quad Z_3^- = \left(0/0/-r'/0/\dots\right), \quad Z_3^+ = \left(0/0/r'/0/\dots\right), \text{ and so forth.}$$

For $s \in \{2, \dots, d\}$ and $v \in \{-1, 1\}^d$ set $E^-_{s,v} = Z_s^- + \eta v$, $E^+_{s,v} = Z_s^+ + \eta v$ and let $e^-_{s,v}, e^+_{s,v} \in \mathbb{R}^d$ be arbitrary elements of $U_\varepsilon(E^-_{s,v})$ and $U_\varepsilon(E^+_{s,v})$, respectively. For $i \in \{1, \dots, 2N'-1\}$ let $x_i \in \mathbb{R}^d$ be an arbitrary element of $U_\delta((-1+\frac{i}{N'}/0/\dots/0))$. Let $\varphi : \{A, B\} \cup \{e^-_{s,v}, e^+_{s,v} : s \in \{2, \dots, d\}, v \in \{-1, 1\}^d\} \cup \{x_i : i = 1, \dots, 2N'-1\} \to \mathbb{R}^d$ be an isotonic function with $\|\varphi(A) - \varphi(B)\| = 2$. Then there exist a constant C depending only on d and an isometry $S : \mathbb{R}^d \to \mathbb{R}^d$ such that

$$\|A - S(\varphi(A))\| \leq C\sqrt{A(\omega')}, \qquad \|B - S(\varphi(B))\| \leq C\sqrt{A(\omega')},$$
$$\|Z_s^m - S(\varphi(e^m_{s,\underline{v}}))\| \leq C\sqrt{A(\omega')}, \quad m \in \{-, +\}, s \in \{2, \dots, d\},$$

where $\underline{v} = (1/1/1/\dots/1)$ and $A(\omega')$ only depends on ω' and d and satisfies $A(\omega') \to 0$ as $\omega' \to 0$.

Proof By composing an isometry with φ, we may assume that $\varphi(A) = A$ and $\varphi(B) = B$. It is straightforward to see that (we set $x_0 = A$ and $x_{2N'} = B$)

$$\max\{\|A - x_1\|, \|B - x_{2N'-1}\|\} < \min\{\|x_j - x_{j'}\| : j \neq j' \in \{0, 2, 4, \dots, 2N'\}\}$$

and $\varphi(x_i) \in [-2, 2]^d$, $i = 0, \dots, 2N'$. With an argument similar to the one subsequent to (2.7) in the proof of Lemma 2.8 we can show that $\|\varphi(A) - \varphi(x_1)\| < \omega'/2$ and $\|\varphi(B) - \varphi(x_{2N'-1})\| < \omega'/2$. Since φ is isotonic, this implies that $\|\varphi(u_1) - \varphi(u_2)\| < \omega'/2$ for all points u_1, u_2 in the domain of φ with $\|u_1 - u_2\| < \|A - x_1\|$ and $\|\varphi(u_1) - \varphi(u_2)\| > 2 - \omega'/2$ for all points u_1, u_2 in the domain of φ with $\|u_1 - u_2\| > \|A - x_{2N'-1}\|$.

The parameters η, δ, and ε are chosen in such a way that $\|e^m_{s,v_1} - e^m_{s,v_2}\| < \|A - x_1\|$ for any $m \in \{-, +\}$, $s \in \{2, \dots, d\}$, and for all $v_1, v_2 \in \{-1, 1\}^d$. Furthermore, we have $\|A - x_{2N'-1}\| < \|e^+_{s,v} - e^-_{s,\hat{v}}\|$ for any $s \in \{2, \dots, d\}$ and for all $v, \hat{v} \in \{-1, 1\}^d$ with v equaling $\hat{v}$ except that $v_s = +1$, $\hat{v}_s = -1$. Using this and the assumption of φ being isotonic, we can show that

$$\begin{aligned} &2 - \omega' < \|p_s^+ - p_s^-\| \le 2, \qquad s = 1, \dots, d, \\ &\|p_s^+ - p_{s'}^-\| - \omega' < \|p_s^+ - p_{s'}^+\| < \|p_s^+ - p_{s'}^-\| + \omega', \quad s \neq s' \in \{1, \dots, d\}, \\ &\|p_s^- - p_{s'}^-\| - \omega' < \|p_s^- - p_{s'}^+\| < \|p_s^- - p_{s'}^-\| + \omega', \quad s \neq s' \in \{1, \dots, d\}, \end{aligned} \tag{2.9}$$

where $p_1^+ = \varphi(B)$, $p_1^- = \varphi(A)$, and $p_s^+ = \varphi(e^+_{s,\underline{v}})$, $p_s^- = \varphi(e^-_{s,\underline{v}})$ for $s = 2, \dots, d$. For example, let us prove that $\|p_1^- - p_2^-\| - \omega' < \|p_1^- - p_2^+\| < \|p_1^- - p_2^-\| + \omega'$: Elementary calculations show that $\|A - e^-_{2,\underline{v}}\| < \|A - e^+_{2,\underline{v}}\|$ and $\|A - e^+_{2,v^c}\| < \|A - e^-_{2,v^c}\|$ with $v^c = (-1/-1/\dots/-1)$. We infer $\|p_1^- - p_2^-\| < \|p_1^- - p_2^+\|$ and $\|p_1^- - \varphi(e^+_{2,v^c})\| < \|p_1^- - \varphi(e^-_{2,v^c})\|$ and thus obtain

$$\begin{aligned} \|p_1^- - p_2^+\| &\le \|p_1^- - \varphi(e^+_{2,v^c})\| + \|\varphi(e^+_{2,v^c}) - p_2^+\| < \|p_1^- - \varphi(e^-_{2,v^c})\| + \|\varphi(e^+_{2,v^c}) - p_2^+\| \\ &\le \|p_1^- - p_2^-\| + \|p_2^- - \varphi(e^-_{2,v^c})\| + \|\varphi(e^+_{2,v^c}) - p_2^+\| < \|p_1^- - p_2^-\| + \omega'. \end{aligned}$$

Using that $\omega' < 1$ and $\|p_s^+ - p_{s'}^-\| < 2$ due to $\|e^+_{s,\underline{v}} - e^-_{s',\underline{v}}\| < \|A - B\|$, we can infer from (2.9) that

$$|\langle p_s^+ - p_s^-, p_{s'}^+ - p_{s'}^- \rangle| < 10\omega', \quad s \neq s' \in \{1, \dots, d\}. \tag{2.10}$$

Furthermore, we can show that $\|(p_s^+ + p_s^-) - (p_{s'}^+ + p_{s'}^-)\|$, $s \neq s' \in \{1, \dots, d\}$, is small provided ω' is small, that is

$$\|(p_s^+ + p_s^-) - (p_{s'}^+ + p_{s'}^-)\|^2 \le d \left(\frac{20\omega' + 8\frac{10d\omega'}{4d-1}(4d)^{d-1}}{2 - \omega' - \frac{10d\omega'}{4d-1}(4d)^{d-1}} \right)^2, \quad s \neq s' \in \{1, \dots, d\}. \tag{2.11}$$

In order to show (2.11), we first apply the Gram-Schmidt process to the vectors $(p_s^+ - p_s^-)$, $s = 1, \dots, d$, that is we define an orthonormal basis of $\mathbb{R}^d$ comprising basis vectors $g_1, \dots, g_d$ defined by $g_1^* = p_1^+ - p_1^-$, $g_1 = g_1^*/\|g_1^*\|$, and

$$g_s^* = (p_s^+ - p_s^-) - \sum_{j=1}^{s-1} \langle p_s^+ - p_s^-, g_j \rangle g_j, \quad g_s = \frac{g_s^*}{\|g_s^*\|}, \quad s = 2, \dots, d.$$

Making use of (2.10) it is elementary to show that $g_1, \ldots, g_d$ are indeed well-defined and that for $s = 1, \ldots, d$

$$\|g_s^* - (p_s^+ - p_s^-)\| \leq \frac{10d\omega'}{4d-1}(4d)^{d-1} \quad \text{and} \quad \left\| g_s - \frac{p_s^+ - p_s^-}{p_s^+ - p_s^-} \right\| \leq \frac{2}{2-\omega'} \frac{10d\omega'}{4d-1}(4d)^{d-1}. \tag{2.12}$$

We now can write $(p_s^+ + p_s^-) - (p_{s'}^+ + p_{s'}^-)$ as

$$(p_s^+ + p_s^-) - (p_{s'}^+ + p_{s'}^-) = \sum_{j=1}^{d} \langle (p_s^+ + p_s^-) - (p_{s'}^+ + p_{s'}^-), g_j \rangle g_j.$$

Using (2.9) and (2.12) we can upper bound the magnitude of the Fourier coefficients by

$$|\langle (p_s^+ + p_s^-) - (p_{s'}^+ + p_{s'}^-), g_j \rangle| \leq \frac{20\omega' + 8\frac{10d\omega'}{4d-1}(4d)^{d-1}}{2 - \omega' - \frac{10d\omega'}{4d-1}(4d)^{d-1}},$$

which immediately yields (2.11).

Now, setting $Z_1^- = A$, $Z_1^+ = B$, we consider the map $f : \{Z_1^-, Z_1^+, \ldots, Z_d^-, Z_d^+\} \to \{p_1^-, p_1^+, \ldots, p_d^-, p_d^+\}$ given by $f(Z_s^m) = p_s^m$ for $m \in \{-, +\}$, $s \in \{1, \ldots, d\}$. Using (2.10) and (2.11) it is straightforward to show that f is a $2\sqrt{A(\omega')}$-nearisometry, that is it holds for all $x, y \in \{Z_1^-, Z_1^+, \ldots, Z_d^-, Z_d^+\}$ that

$$\|x - y\| - 2\sqrt{A(\omega')} \leq \|f(x) - f(y)\| \leq \|x - y\| + 2\sqrt{A(\omega')},$$

where

$$A(\omega') = \frac{1}{4} d \left(\frac{20\omega' + 8\frac{10d\omega'}{4d-1}(4d)^{d-1}}{2 - \omega' - \frac{10d\omega'}{4d-1}(4d)^{d-1}} \right)^2 + 2\sqrt{d}\, \frac{20\omega' + 8\frac{10d\omega'}{4d-1}(4d)^{d-1}}{2 - \omega' - \frac{10d\omega'}{4d-1}(4d)^{d-1}} + 5\omega'.$$

According to Alestalo et al. (2001), Theorem 3.3, there exists a constant C' (depending only on d—we can choose it independently of N', the parameters $r', \eta, \delta, \varepsilon$, and the precise locations of $e_{s,v}^-, e_{s,v}^+$, and x_i) and an isometry $T : \mathbb{R}^d \to \mathbb{R}^d$ such that $\|T(x) - f(x)\| \leq 2C'\sqrt{A(\omega')}$, $x \in \{Z_1^-, Z_1^+, \ldots, Z_d^-, Z_d^+\}$. We set $S = T^{-1}$ and $C = 2C'$, and the assertion of Lemma 2.9 follows immediately. ■

Now we can prove Theorem 2.3 for the case $d \geq 2$.

Proof of Part 1 of Theorem 2.3: By Lemma 2.16 (see Section 2.6.1) it is sufficient to prove that for every $\varepsilon_0 > 0$ there exists $N(\varepsilon_0) \in \mathbb{N}$ such that for all $n \geq N(\varepsilon_0)$ there is a similarity transformation $S(n, \varepsilon_0) : \mathbb{R}^d \to \mathbb{R}^d$ with $\|\varphi_n - S(n, \varepsilon_0)\|_{\infty(\{x_1, \ldots, x_n\})} < \varepsilon_0$.

In a nutshell, the basic idea for proving this is the following: Assume K is a ball with diameter only slightly larger than two and containing all the balls of Lemma 2.8. If $n \in \mathbb{N}$ is sufficiently large, in each of these balls there is an element of $\{x_1, \ldots, x_n\}$. Assume for the moment that φ_n satisfies Assumption ($\sharp$) of Lemma 2.8. Then we obtain from (2.5) an estimate for the expression $\|\varphi_n(x) - \varphi_n(y)\|$ for roughly uniformly distributed values of $\|x - y\|$ in $[0, 2] \approx [0, \operatorname{diam}\{x_1, \ldots, x_n\}]$. Since φ_n is isotonic, this gives an estimate for $\|\varphi_n(x) - \varphi_n(y)\|$ for all $x, y \in \{x_1, \ldots, x_n\}$ that is sufficient to show that φ_n is an ε-nearisometry for some small ε. Hence, we can uniformly approximate φ_n by an isometry according to Alestalo et al. (2001). It remains to be argued why Assumption ($\sharp$) of Lemma 2.8 indeed can be taken as satisfied. However, this is the statement of Lemma 2.9.

In detail, the main steps of the proof are as follows:

1. By composing the functions φ_n with a similarity transformation we may assume that $K = B_1(0)$.

2. - Denote by $\widetilde{C}$ the constant from Theorem 2.2 in Alestalo et al. (2001), which only depends on d. Choose $N^* \in \mathbb{N}$ sufficiently large and $\tilde{\varepsilon} > 0$ sufficiently small such that
$$4R\widetilde{C}\sqrt{\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}} < \varepsilon_0.$$
   - Choose $N \in \mathbb{N}$ and the parameters $\alpha, \tilde{\alpha} \in \mathbb{R}^d$, $\mu > 0$, $(\varepsilon_k)_{1 \le k \le N}$, and $(\delta_k)_{1 \le k \le N}$ in Lemma 2.8 such that all assumptions of Lemma 2.8 (see the detailed version in Section 2.6.3) are satisfied for any choice of the parameters $r, \tilde{r} \in \mathbb{R}^d$ with $1/2 \le r_i = \tilde{r}_i \le 1$, $i = 1, \ldots, d$, and with N^* as chosen above, and such that $\omega = \omega(N)$ satisfies $3d\sqrt{\omega} < \tilde{\varepsilon}$.
   - Choose $N' \in \mathbb{N}$ such that the expression $C\sqrt{A(\omega')}$ from Lemma 2.9 satisfies $C\sqrt{A(\omega')} < \min_{i=1,\ldots,d} \tilde{\alpha}_i$. Choose the parameters $r', \eta, \delta, \varepsilon > 0$ in Lemma 2.9 such that all assumptions of Lemma 2.9 (see the detailed version in Section 2.6.3) are satisfied and such that $\eta\sqrt{d} + \varepsilon < \min_{i=1,\ldots,d} \alpha_i$. Choose $r, \tilde{r} \in \mathbb{R}^d$ in Lemma 2.8 as $r = \tilde{r} = (1/r'/r'/\ldots/r')$.
   - Let $\tau > 0$ such that $\frac{2}{2-2\tau} < \frac{2^{N^*}+3}{2^{N^*}}$ and $0 < \tau' < \frac{\tau}{2}$ such that
$$U_{\frac{\|u_A - u_B\|}{2}}\left(\frac{u_A + u_B}{2}\right) \subseteq K$$
holds for all $(u_A, u_B) \in U_{\tau'}((-1 + \frac{\tau}{2}/0/0/\ldots/0)) \times U_{\tau'}((1 - \frac{\tau}{2}/0/0/\ldots/0))$. Using a continuity argument it is easy to see that such τ' actually exists.
   - Let $r_{\min} > 0$ be the minimal radius of the finitely many open balls defined in Lemma 2.8 and Lemma 2.9 with the specified parameters. Choose $0 < \nu < \min\{\tau', \frac{r_{\min}}{2}, \frac{\tilde{\varepsilon}}{8}\}$. Since $\{x_n : n \in \mathbb{N}\}$ is dense in K and K is compact, there exists $N_0 \in \mathbb{N}$ such that
$$\forall y \in K : U_\nu(y) \cap \{x_1, \ldots, x_{N_0}\} \neq \emptyset. \tag{2.13}$$
We set $N(\varepsilon_0) = N_0$ and consider $\varphi_l : \{x_1, \ldots, x_l\} \to U_R(0)$ for $l \ge N(\varepsilon_0)$.

3. We can choose points $x_A, x_B \in \{x_1, \ldots, x_l\}$ with $x_A \in U_{\tau'}((-1 + \frac{\tau}{2}/0/0/\ldots/0))$ and $x_B \in U_{\tau'}((1 - \frac{\tau}{2}/0/0/\ldots/0))$, respectively. Let $T : \mathbb{R}^d \to \mathbb{R}^d$ be a similarity transformation satisfying $T(x_A) = (-1/0/0/\ldots/0)$ and $T(x_B) = (1/0/0/\ldots/0)$. For its scale factor $\lambda(T) = \|T(x_A) - T(x_B)\| / \|x_A - x_B\|$ we have

$$1 < \frac{2}{2 - \tau + 2\tau'} < \lambda(T) < \frac{2}{2 - \tau - 2\tau'} < \frac{2}{2 - 2\tau} < \frac{2^{N^*} + 3}{2^{N^*}} < 2. \tag{2.14}$$

   Due to $U_{\|x_A - x_B\|/2}((x_A + x_B)/2) \subseteq K$ we have $U_1(0) \subseteq T(K)$.

4. We have $U_{2\nu}(a) \cap \{T(x_1), \ldots, T(x_l)\} \neq \emptyset$ for any $a \in T(K)$ due to $T^{-1}(U_{2\nu}(a)) = U_{\frac{1}{\lambda(T)}2\nu}(T^{-1}(a)) \supseteq U_{\frac{2\nu}{2}}(T^{-1}(a)) = U_\nu(T^{-1}(a))$ and (2.13). In particular, $U_{r_{\min}}(a) \cap \{T(x_1), \ldots, T(x_l)\} \neq \emptyset$ for $a \in T(K)$, and because of $U_1(0) \subseteq T(K)$ in every open ball defined in Lemma 2.8 and Lemma 2.9 with the specified parameters there is an element of $\{T(x_1), \ldots, T(x_l)\}$.

5. Let $U : \mathbb{R}^d \to \mathbb{R}^d$ be a similarity transformation with $\|U(\varphi_l(x_A)) - U(\varphi_l(x_B))\| = 2$. For its scale factor $\lambda(U)$ we have

$$\lambda(U) = \frac{\|U(\varphi_l(x_A)) - U(\varphi_l(x_B))\|}{\|\varphi_l(x_A) - \varphi_l(x_B)\|} > \frac{2}{2R} = \frac{1}{R}. \tag{2.15}$$

6. We consider the isotonic function $U \circ \varphi_l \circ T^{-1} : \{T(x_1), \ldots, T(x_l)\} \to \mathbb{R}^d$. In every open ball defined in Lemma 2.9 there is an element of $\{T(x_1), \ldots, T(x_l)\}$. We denote these elements as in Lemma 2.9 ($A = T(x_A), B = T(x_B)$). According to Lemma 2.9 there exists an isometry $S : \mathbb{R}^d \to \mathbb{R}^d$ such that

$$\begin{aligned} &\|A - S(U \circ \varphi_l \circ T^{-1}(A))\| \leq C\sqrt{A(\omega')} < \tilde{\alpha}_1, \quad \|B - S(U \circ \varphi_l \circ T^{-1}(B))\| < \tilde{\alpha}_1, \\ &\|Z_s^m - S(U \circ \varphi_l \circ T^{-1}(e_{s,\underline{v}}^m))\| \leq C\sqrt{A(\omega')} < \tilde{\alpha}_s, \quad m \in \{-,+\}, s \in \{2, \ldots, d\}. \end{aligned}$$

7. We consider the isotonic function $S \circ U \circ \varphi_n \circ T^{-1} : \{T(x_1), \ldots, T(x_l)\} \to \mathbb{R}^d$. In every open ball of Lemma 2.8 there is an element of $\{T(x_1), \ldots, T(x_l)\}$. We denote these elements as in Lemma 2.8 ($X_1^- = A = T(x_A), X_1^+ = B = T(x_B), X_s^+ = e_{s,\underline{v}}^+, X_s^- = e_{s,\underline{v}}^-$). Due to $\eta\sqrt{d} + \varepsilon < \min_{i=1,\ldots,d} \alpha_i$ we have $e_{s,\underline{v}}^{\tilde{m}} \in U_{\alpha_s}(m_s^{\tilde{m}})$, $\tilde{m} \in \{-,+\}$, $s \in \{2, \ldots, d\}$, and according to the previous step Assumption ($\sharp$) of Lemma 2.8 is satisfied. Hence,

$$\|y_{k,i}^m - S \circ U \circ \varphi_n \circ T^{-1}(y_{k,i}^m)\| < 3d\sqrt{\omega} < \tilde{\varepsilon}$$

   for $k \leq N^*$, $i \in \{1, 3, \ldots, 2^k - 1\}$ and $m \in \{l, r\}$.

8. We show that $f = S \circ U \circ \varphi_n \circ T^{-1}$ is a $\left(\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}\right)$-nearisometry, that is f satisfies

$$\|x - y\| - \left(\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}\right) \leq \|f(x) - f(y)\| \leq \|x - y\| + \left(\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}\right) \tag{2.16}$$

for all $x, y \in \{T(x_1), \ldots, T(x_l)\}$. For elements x, y with $\|x - y\| \leq (2^{N^*} - 2)/2^{N^*-1}$ it is straightforward to prove the even tighter estimate

$$\|x-y\| - \left(\frac{2}{2^{N^*-1}} + 2\tilde{\varepsilon}\right) \leq \|f(x) - f(y)\| \leq \|x-y\| + \left(\frac{2}{2^{N^*-1}} + 2\tilde{\varepsilon}\right) \tag{2.17}$$

by approximating $\|x-y\|$ by $\|\tilde{x} - \tilde{y}\|$ with elements $\tilde{x}, \tilde{y} \in \{y_{k,i}^m : m \in \{l, r\}, k \leq N^*, i \in \{1, 3, \ldots, 2^k - 1\}\}$ and using that f is isotonic.

For elements x, y with $(2^{N^*} - 2)/2^{N^*-1} < \|x-y\| \leq \operatorname{diam} T(K)$ set $x' = x + \frac{1}{3}(y-x)$ and $y' = x + \frac{2}{3}(y-x)$. Since K is assumed to be convex, so is $T(K)$ and hence $x', y' \in T(K)$. Let $x^a \in U_{\tilde{\varepsilon}/4}(x') \cap \{T(x_1), \ldots, T(x_l)\}$ and $y^a \in U_{\tilde{\varepsilon}/4}(y') \cap \{T(x_1), \ldots, T(x_l)\}$ (such elements exist according to Step 4). We can approximate $\|x-y\| = \|x-x'\| + \|x'-y'\| + \|y'-y\|$ by $\|x-x^a\| + \|x^a - y^a\| + \|y^a - y\|$ where each summand is smaller than $(2^{N^*} - 2)/2^{N^*-1}$. Using (2.17) and $\operatorname{diam} T(K) = 2\lambda(T) < (2^{N^*} + 3)/2^{N^*-1}$ according to (2.14), it is not hard to show (2.16).

9. We have $\operatorname{diam}(\{T(x_1), \ldots, T(x_l)\}) \geq 2$ because of $A, B \in \{T(x_1), \ldots, T(x_l)\}$. According to Alestalo et al. (2001), Theorem 2.2, there exists an isometry $S' : \mathbb{R}^d \to \mathbb{R}^d$ such that

$$\begin{aligned}\|S' - S \circ U \circ \varphi_n \circ T^{-1}\|_{\infty(\{T(x_1), \ldots, T(x_l)\})} \leq & \widetilde{C} \cdot \operatorname{diam}(\{T(x_1), \ldots, T(x_l)\}) \cdot \\ & \sqrt{\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}}.\end{aligned}$$

Making use of (2.15) and $\operatorname{diam}(\{T(x_1), \ldots, T(x_l)\}) = \lambda(T) \cdot \operatorname{diam}(\{x_1, \ldots, x_l\}) < 4$ according to (2.14), this implies that

$$\begin{aligned}\|U^{-1} \circ S^{-1} \circ S' \circ T - \varphi_n\|_{\infty(\{x_1, \ldots, x_l\})} &< \frac{1}{\lambda(U)} 4\widetilde{C} \sqrt{\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}} \\ &< R4\widetilde{C} \sqrt{\frac{6}{2^{N^*-1}} + 7\tilde{\varepsilon}} < \varepsilon_0.\end{aligned}$$

■

Proof of Part 2 of Theorem 2.3: Since $\cup_{i=1}^k K_i^\circ$ is assumed to be connected, we can write K as $K = \cup_{i=1}^{k'} K_i'$ with $K_i' \in \{K_1, \ldots, K_k\}$ and such that $K_i'^\circ \cap K_{i+1}'^\circ \neq \emptyset$, $i = 1, \ldots, k'-1$. Hence, w.l.o.g. we may assume that $K = \cup_{i=1}^k K_i$ such that $K_i^\circ \cap K_{i+1}^\circ \neq \emptyset$, $i = 1, \ldots, k-1$.

For every $i \in \{1, \ldots, k\}$ it holds that $\{x_n : n \in \mathbb{N}\} \cap K_i$ is dense in K_i because of $K_i \subseteq \overline{K_i^\circ}$, and hence there exists a sequence $(S_n^i)_{n \in \mathbb{N}}$ of similarity transformations such that

$$\|S_n^i - \varphi_n|_{\{x_1, \ldots, x_n\} \cap K_i}\|_{\infty(\{x_1, \ldots, x_n\} \cap K_i)} \to 0. \tag{2.18}$$

We prove that

$$\|S_n^1 - \varphi_n|_{\{x_1, \ldots, x_n\} \cap (K_1 \cup \ldots \cup K_j)}\|_{\infty(\{x_1, \ldots, x_n\} \cap (K_1 \cup \ldots \cup K_j))} \to 0, \quad j = 1, \ldots, k, \tag{2.19}$$

by induction over j. The basis is clear. For the inductive step assume that (2.19) holds for some $j < k$. We have to infer that it also holds for $j+1$. So let $\varepsilon > 0$ be arbitrary.

Since $K_j^\circ \cap K_{j+1}^\circ \neq \emptyset$, we can choose $\widehat{N} \in \mathbb{N}$ such that $\{x_1, \ldots, x_{\widehat{N}}\}$ contains $d+1$ affinely independent points in $K_j \cap K_{j+1}$. Denote these points by $u_1, u_2, \ldots, u_{d+1}$. Any point $u \in \mathbb{R}^d$ can be written as $u = \sum_{i=1}^{d+1} \lambda_i(u) u_i$ for some unique coefficients $\lambda_i(u) \in \mathbb{R}$ with $\sum_{i=1}^{d+1} \lambda_i(u) = 1$. Since K is bounded, there exists $C > 0$ such that $|\lambda_i(u)| \leq C$, $u \in K$, $i \in \{1, \ldots, d+1\}$. Choose $\tilde{\varepsilon} > 0$ such that $2(d+1)C\tilde{\varepsilon} + \tilde{\varepsilon} < \varepsilon$. Choose $N_1 \in \mathbb{N}$ such that

$$\|S_n^1 - \varphi_n|_{\{x_1,\ldots,x_n\}\cap(K_1\cup\ldots\cup K_j)}\|_{\infty(\{x_1,\ldots,x_n\}\cap(K_1\cup\ldots\cup K_j))} < \tilde{\varepsilon}, \quad n \geq N_1,$$

which is possible because of the induction hypothesis. Choose $N_2 \in \mathbb{N}$ such that

$$\|S_n^{j+1} - \varphi_n|_{\{x_1,\ldots,x_n\}\cap K_{j+1}}\|_{\infty(\{x_1,\ldots,x_n\}\cap K_{j+1})} < \tilde{\varepsilon}, \quad n \geq N_2,$$

which is possible due to (2.18). For $n \geq \max\{\widehat{N}, N_1, N_2\}$ and $x \in \{x_1, \ldots, x_n\} \cap (K_1 \cup \ldots \cup K_{j+1})$ we then have:

- If $x \in K_1 \cup \ldots \cup K_j$, then clearly $\|S_n^1(x) - \varphi_n(x)\| < \tilde{\varepsilon} < \varepsilon$.
- If $x \in K_{j+1}$, then $\|S_n^1(x) - \varphi_n(x)\| \leq \|S_n^1(x) - S_n^{j+1}(x)\| + \|S_n^{j+1}(x) - \varphi_n(x)\| < \|S_n^1(x) - S_n^{j+1}(x)\| + \tilde{\varepsilon}$. Since $x = \sum_{i=1}^{d+1} \lambda_i u_i$ with $\sum_{i=1}^{d+1} \lambda_i = 1$ and $|\lambda_i| \leq C$, $i = 1, \ldots, d+1$, we have

$$\begin{aligned}\|S_n^1(x) - S_n^{j+1}(x)\| &= \left\|\sum_{i=1}^{d+1} \lambda_i S_n^1(u_i) - \sum_{i=1}^{d+1} \lambda_i S_n^{j+1}(u_i)\right\| \\ &\leq \sum_{i=1}^{d+1} |\lambda_i| \|S_n^1(u_i) - S_n^{j+1}(u_i)\| \\ &\leq (d+1)C \max_{i=1,\ldots,d+1} \|S_n^1(u_i) - S_n^{j+1}(u_i)\|.\end{aligned}$$

  Since $\|S_n^1(u_i) - S_n^{j+1}(u_i)\| \leq \|S_n^1(u_i) - \varphi_n(u_i)\| + \|\varphi_n(u_i) - S_n^{j+1}(u_i)\| < 2\tilde{\varepsilon}$ for $i \in \{1, \ldots, d+1\}$, this yields $\|S_n^1(x) - \varphi_n(x)\| \leq 2(d+1)C\tilde{\varepsilon} + \tilde{\varepsilon} < \varepsilon$.

■

2.4 Proof of Theorem 2.4 (the infinite case)

The proof of Theorem 2.4 consists of a number of steps, which we formulate as separate lemmas and propositions.

Lemma 2.10 (Isotonic implies continuous)**.** *Let $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ and $f : \Omega \to \mathbb{R}^d$ be a locally isotonic function. Then f is continuous. If we additionally assume Ω to be a set with at least one limit point that is contained in it and f to be globally isotonic, then f is even uniformly continuous.*

Proof Since continuity is a local property, for the first part of the lemma it suffices to show that for any point $x \in \Omega$ there is a neigborhood $U(x)$ in Ω such that $f|_{U(x)}$ is continuous. Hence, w.l.o.g. we may assume f to be globally isotonic. The key observation is that if f was discontinuous at one point, the distance between different points in $f(\Omega)$ would be bounded from below by a positive constant: Assume there was $x_0 \in \Omega$ such that f was discontinuous at x_0, that is

$$\exists \varepsilon > 0 \;\; \forall \delta > 0 \;\; \exists x \in \Omega : \|x_0 - x\| < \delta \quad \wedge \quad \|f(x_0) - f(x)\| \geq \varepsilon.$$

Since f is isotonic, this implies

$$\forall x \neq y \in \Omega : \|f(x) - f(y)\| \geq \varepsilon. \tag{2.20}$$

In case that Ω is uncountable, this immediately contradicts the separability of $\mathbb{R}^d$. In general, a compactness argument leads to a contradiction: Let $r > 0$ such that there exist $\tilde{x}, \tilde{y} \in \Omega$ satisfying $\|\tilde{x} - \tilde{y}\| > 2r$ and a closed ball $B_r(\tilde{z}) \subseteq \mathbb{R}^d$ such that $B_r(\tilde{z}) \cap \Omega$ contains infinitely many points. Note that such $r > 0$ surely exists since x_0 is a limit point. For all $x, y \in B_r(\tilde{z}) \cap \Omega$ we have $\|x - y\| \leq 2r < \|\tilde{x} - \tilde{y}\|$ and hence $\|f(x) - f(y)\| < \|f(\tilde{x}) - f(\tilde{y})\|$. It follows that $f(B_r(\tilde{z}) \cap \Omega)$ is bounded and $\overline{f(B_r(\tilde{z}) \cap \Omega)}$ is compact in $\mathbb{R}^d$. Consider the family of open balls $U_\varepsilon(f(x))$, $x \in B_r(\tilde{z}) \cap \Omega$, which covers $\overline{f(B_r(\tilde{z}) \cap \Omega)}$. As a consequence, there exist $x_1, \ldots, x_n \in B_r(\tilde{z}) \cap \Omega$ such that $U_\varepsilon(f(x_1)), \ldots, U_\varepsilon(f(x_n))$ cover $\overline{f(B_r(\tilde{z}) \cap \Omega)}$. Hence, if $x \in B_r(\tilde{z}) \cap \Omega$, we have $f(x) \in U_\varepsilon(f(x_i))$ for some $i \in \{1, \ldots, n\}$. Choosing $x \in (B_r(\tilde{z}) \cap \Omega) \backslash \{x_1, \ldots, x_n\}$ yields a contradiction to (2.20).

In order to prove the second claim, let $x_0 \in \Omega$ be a limit point of Ω and let $\varepsilon > 0$ be arbitrary. We already know that f is continuous, and hence there exists $\delta > 0$ such that $\|f(x) - f(x_0)\| < \varepsilon$ for all $x \in \Omega$ with $\|x - x_0\| < \delta$. Let $x' \in \Omega$ with $0 < \|x' - x_0\| = \delta' < \delta$ (since x_0 is a limit point, there is such a point x'). For all $x, y \in \Omega$ with $\|x - y\| < \delta'$ we have $\|x - y\| < \|x' - x_0\|$ and hence $\|f(x) - f(y)\| < \|f(x') - f(x_0)\| < \varepsilon$. ■

The next lemma shows that if $\Omega \subseteq \mathbb{R}^d$ is a ball and $f : \Omega \to \mathbb{R}^d$ is weakly isotonic, then f is even strongly isotonic, at least on a slightly smaller ball.

Lemma 2.11 (Weakly isotonic implies strongly isotonic on balls)**.** *Let $\Omega = U_\varepsilon(z) \subseteq \mathbb{R}^d$ for some arbitrary $\varepsilon > 0$ and $z \in \mathbb{R}^d$. Let $f : \Omega \to \mathbb{R}^d$ be weakly isotonic. Then $f|_{U_{\varepsilon/4}(z)}$ is strongly isotonic.*

Proof Let $x \neq y, v \neq w \in U_{\varepsilon/4}(z)$ with $\|x - y\| < \|v - w\|$ be arbitrary. In order to prove that $f|_{U_{\varepsilon/4}(z)}$ is isotonic, we have to show that $\|f(x) - f(y)\| < \|f(v) - f(w)\|$. We first consider the case that $\|w - y\| \leq \|x - y\|$. Let $r \in \mathbb{R}^d$ with $\|r\| = 1$ and $\langle r, w - y \rangle = 0$, that is r is orthogonal to $w - y$. For $\alpha \in \mathbb{R}$ set

$$u(\alpha) = y + \left(\frac{\|y - w\|}{2} - \alpha \right) \frac{w - y}{\|w - y\|} + \beta r$$

for some fixed $\sqrt{\|x - y\|^2 - \frac{\|w-y\|^2}{4}} < \beta < \sqrt{\|w - v\|^2 - \frac{\|w-y\|^2}{4}}$. It is easy to check that $u(0) \in \Omega$ and that $\|x - y\| < \|y - u(0)\|$ and $\|u(0) - w\| < \|w - v\|$. Clearly, $u(\alpha)$

continuously depends on α, and hence there also exists some small $\alpha_0 > 0$ such that $u(\alpha_0) \in \Omega$ and $\|x-y\| < \|y-u(\alpha_0)\|$ and $\|u(\alpha_0)-w\| < \|w-v\|$. However, for $\alpha > 0$ we have $\|u(\alpha)-y\| < \|u(\alpha)-w\|$ and hence

$$\|x-y\| < \|y-u(\alpha_0)\| < \|u(\alpha_0)-w\| < \|w-v\|.$$

Since f is weakly isotonic on Ω, this implies that

$$\|f(x)-f(y)\| < \|f(y)-f(u(\alpha_0))\| < \|f(u(\alpha_0))-f(w)\| < \|f(w)-f(v)\|.$$

Now assume that $\|x-y\| < \|w-y\|$. It is easy to see that we can choose a finite sequence of pairs of points $(x_i, y_i)_{i=1,\dots,n}$ such that all these points are located on the line segment connecting y and w and such that the following holds:

$$\begin{aligned}
&\|x-y\| < \|x_1-y_1\|, \qquad \|x_1-y\| \le \|x-y\|,\\
&\|x_{i-1}-y_{i-1}\| < \|x_i-y_i\|, \qquad \|x_i-y_{i-1}\| \le \|x_{i-1}-y_{i-1}\|, \quad i=2,\dots,n,\\
&\|x_n-y_n\| < \|w-v\|, \qquad \|y_n-w\| \le \|y_n-x_n\|.
\end{aligned}$$

With the same argument as above we can show that

$$\|f(x)-f(y)\| < \|f(x_1)-f(y_1)\| < \ldots < \|f(x_n)-f(y_n)\| < \|f(w)-f(v)\|.$$

We use continuity of $f|_{U_{\varepsilon/4}(z)}$ (compare with Lemma 2.10) in order to show that it is even strongly isotonic: Let $x \neq y, v \neq w \in U_{\varepsilon/4}(z)$ with $\|x-y\| = \|v-w\|$. We can choose sequences $x_n \to x$ and $y_n \to y$ such that all points x_n, y_n are located in the interior of the line segment connecting x and y. It follows that $\|x_n-y_n\| < \|v-w\|$ and hence $\|f(x_n)-f(y_n)\| < \|f(z)-f(w)\|$. Taking the limit $n \to \infty$ yields $\|f(x)-f(y)\| \le \|f(v)-f(w)\|$. Similarly, we can show that $\|f(v)-f(w)\| \le \|f(x)-f(y)\|$ and hence $\|f(x)-f(y)\| = \|f(v)-f(w)\|$. ■

The following proposition already shows that for functions defined on an open and connected set all the properties that we defined in Definition 2.2 are equivalent. The key ingredient in the proof is that the midpoint of a line segment between two points is mapped by a strong isotony to the midpoint of the line segment between the corresponding image points.

Proposition 2.12 (Weakly isotonic implies similarity). *Let $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ be open and connected and $f : \Omega \to \mathbb{R}^d$ be a locally weakly isotonic function. Then f is globally a similarity.*

Proof (details can be found in Section 2.6.2) First, we consider a globally strongly isotonic function $f : \Omega = B_r(z) \to \mathbb{R}^d$, where $r > 0$ and $z \in \mathbb{R}^d$ are arbitrary. This allows us to define a function $\mu : [0, \operatorname{diam}\Omega] \to [0, \operatorname{diam} f(\Omega)]$ by $\mu(\|x-y\|) = \|f(x)-f(y)\|$ for all $x, y \in \Omega$. In order to show that f is a similarity, we have to show that μ is linear. By showing that the midpoint of a line segment between two points in Ω is mapped by f to the midpoint of the line segment between the corresponding image points, we iteratively obtain $\mu(\frac{j}{2^i}\operatorname{diam}\Omega) = \frac{j}{2^i}\operatorname{diam} f(\Omega)$, $i \in \mathbb{N}$, $j \in \{0, \dots, 2^i\}$ (see Section 2.6.2

for details). According to Lemma 2.10, f is continuous and so is μ, implying that $\mu(t) = t \cdot (\operatorname{diam} f(\Omega)/\operatorname{diam}\Omega)$.

Now assume that Ω is open and connected and $f : \Omega \to \mathbb{R}^d$ is a locally weakly isotonic function. According to Lemma 2.11, f is locally strongly isotonic. Hence, given $x \in \Omega$ we can choose $\varepsilon(x) > 0$ such that $B_{\varepsilon(x)}(x) \subseteq \Omega$ and that $f|_{B_{\varepsilon(x)}(x)} : B_{\varepsilon(x)}(x) \to \mathbb{R}^d$ is globally strongly isotonic. It follows from the above that $f|_{B_{\varepsilon(x)}(x)}$ is a similarity and $f : \Omega \to \mathbb{R}^d$ is locally a similarity. According to Lemma 2.15 (see Section 2.6.1), f is even globally a similarity. ■

Finally, the following lemma states that a continuous extension of an isotonic mapping is isotonic too.

Lemma 2.13 (Continuous extension inherits isotony)**.** *Let $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ such that $K = \overline{\Omega}$ is convex. Let $f : \Omega \to \mathbb{R}^d$ be isotonic and $F : K \to \mathbb{R}^d$ be a continuous extension of f. Then F is isotonic.*

Proof We have to show that $\|F(x) - F(y)\| < \|F(v) - F(w)\|$ for all $x, y, v, w \in K$ with $\|x - y\| < \|v - w\|$. Approximating x, y, v, and w by appropriate sequences in Ω and using that $f = F|_\Omega$ is isotonic and F is continuous, this is straightforward. ■

We have collected all ingredients to prove Theorem 2.4.

Proof of Theorem 2.4: We first consider the case that f is globally isotonic and $\overline{\Omega} = \overline{G}$ is convex. Since f is uniformly continuous according to Lemma 2.10, there exists a unique continuous extension $\widetilde{F}$ of f to $\overline{\Omega}$. By Lemma 2.13, $\widetilde{F}$ is isotonic. According to Proposition 2.12, $\widetilde{F}|_G$ is even a similarity. By Lemma 2.14 (see Section 2.6.1), $\widetilde{F}|_G$ can be uniquely extended to a similarity transformation $F : \mathbb{R}^d \to \mathbb{R}^d$.

In the general case, for $x \in G$ let $\varepsilon(x) > 0$ such that $B_{\varepsilon(x)}(x) \subseteq G$ and $f|_{\Omega \cap U_{\varepsilon(x)}(x)}$ is isotonic. Since $\overline{\Omega \cap U_{\varepsilon(x)}(x)} = B_{\varepsilon(x)}(x)$, the above shows that for $x \in G$ there exists a similarity $F_x : \mathbb{R}^d \to \mathbb{R}^d$ such that $F_x|_{\Omega \cap U_{\varepsilon(x)}(x)} = f|_{\Omega \cap U_{\varepsilon(x)}(x)}$. If $x \neq y \in G$ with $U_{\varepsilon(x)}(x) \cap U_{\varepsilon(y)}(y) \neq \emptyset$, the similarities F_x and F_y have to coincide on $\Omega \cap U_\delta(z)$ for $U_\delta(z) \subseteq U_{\varepsilon(x)}(x) \cap U_{\varepsilon(y)}(y)$ and according to Lemma 2.14 they coincide on $\mathbb{R}^d$. Using that G is path-connected, we can even show that $F_x = F_y$ for all $x, y \in G$ similarly to the proof of Lemma 2.15 (see Section 2.6.1). Hence, setting $F = F_{x_0}$ for an arbitrary $x_0 \in G$ yields an extension of f to a similarity transformation. It follows from Lemma 2.14 that this extension is unique. ■

2.5 Discussion

In this chapter we have formalized Shepard's long-standing conjecture using the notion of isotonic functions and established the asymptotic uniqueness of ordinal embeddings for Euclidean data sets, upon knowledge of all ordinal constraints $\|A - B\| < \|C - D\|$. We have also shown the uniqueness property to hold true if only ordinal constraints involving points within one of several small overlapping regions have to be preserved (Part 2 of Theorem 2.3).

Our results give rise to a number of follow-up questions: For the case $d = 1$ we have proved the uniqueness property to hold upon knowledge of only similarity triplets, that is ordinal constraints of the form $\|A - B\| < \|A - C\|$ (Proposition 2.7). Is this also possible for the case $d \geq 2$? Can we further reduce the number of ordinal constraints that are required to be preserved and still guarantee asymptotically unique ordinal embeddings? More generally, which types of ordinal data (compare with Section 1.3) require which number of corresponding constraints for the uniqueness property to hold? Our proofs are based on considering special point configurations for which we can explicitly show that all point coordinates are determined by the ordinal constraints $\|A - B\| < \|C - D\|$ up to a similarity transformation and small perturbations. In doing so, we make use of several constraints involving four different points, and it is not clear how we could adapt our proofs if these constraints were not available. Another question is whether we can provide convergence rates in (2.2). It would be desirable to upper bound the approximation error $\|S_n - \varphi_n\|_{\infty(\{x_1,\ldots,x_n\})}$ in terms of a quantity that describes how well the points $x_1, \ldots, x_n$ approximate or "fill up" K, for example, in terms of the Hausdorff distance $\mathrm{d_H}(K, \{x_1, \ldots, x_n\})$ between K and $\{x_1, \ldots, x_n\}$. For the case $d = 1$ we can use Lemma 2.6 to derive an upper bound $\|S_n - \varphi_n\|_{\infty(\{x_1,\ldots,x_n\})} \in \mathcal{O}(\mathrm{d_H}(K, \{x_1, \ldots, x_n\})^{1/(1+\varepsilon)})$ for arbitrarily fixed $\varepsilon > 0$. For the case $d \geq 2$, using Lemma 2.8 and Lemma 2.9, in principle we should be able to derive an error bound too, but here the interdependencies of the parameters are much more involved, and we were not able to resolve them. We also suspect that any bound derived in this way would be rather weak. A further interesting question is whether our results hold true if we consider Euclidean data sets in $\mathbb{R}^d$ and ordinal embeddings in $\mathbb{R}^{d'}$ for $d' > d$ (in general, ordinal embeddings defined via (2.1) will not exist for $d' < d$). Finally, we could ask about the uniqueness of ordinal embeddings when dealing with non-Euclidean data sets that do not permit perfect ordinal embeddings or when given ordinal distance information that is contaminated by noise. Here it is less clear how to formalize the problem since an ordinal embedding cannot preserve all given constraints.

Motivated by our work and partially building up on it, Arias-Castro (2015) has generalized our results and answered some of the addressed questions. He defines an isotonic function $f : \Omega \to \mathbb{R}^d$ for arbitrary $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ by requiring that

$$\|x - y\| < \|z - w\| \Rightarrow \|f(x) - f(y)\| \leq \|f(z) - f(w)\|, \quad x, y, z, w \in \Omega,$$

and a weakly isotonic function by requiring that this property holds for all $x, y, z, w \in \Omega$ with $x = z$. These definitions differ from ours (compare with Definition 2.2) regarding the weak instead of a strict inequality on the right-hand side of the implication. The weak inequality implies that the set of (weakly) isotonic functions is closed under pointwise convergence. This allows Arias-Castro to establish Part 1 of our Theorem 2.3 only assuming the functions φ_n to be weakly isotonic and for more general sets K, for example, bounded, open, and connected sets with a boundary of bounded curvature. His arguments are simpler than ours and rely on the diagonal argument in the proof of the Arzelà–Ascoli theorem to show pointwise convergence of a subsequence of $(\varphi_n)_{n\in\mathbb{N}}$. Furthermore, Arias-Castro proves that $\|S_n - \varphi_n\|_{\infty(\{x_1,\ldots,x_n\})} \in \mathcal{O}(\mathrm{d_H}(K, \{x_1, \ldots, x_n\}))$ if one assumes the functions φ_n to be isotonic and $\|S_n - \varphi_n\|_{\infty(\{x_1,\dots,x_n\})} \in \mathcal{O}(\mathrm{d_H}(K, \{x_1, \dots, x_n\})^{1/2})$ if one only assumes them to be weakly isotonic. The proofs of these assertions look similar to our proofs. In particular, Arias-Castro makes use of the results by Alestalo et al. (2001) too, but he mainly builds on regular simplexes instead of cross-polytopes as we do. Arias-Castro leaves it as an open problem whether the rates that he provided are tight. For isotonic functions and in the one-dimensional case $d = 1$ we can show with a simple example that the provided rate is tight up to constants: Consider $K = [0, 1]$ and for $n \in \mathbb{N}$ the point set $x_0, \dots, x_{n-1}$ given by $x_i = i/n$, $i = 0, \dots, n-1$. The function φ_n that maps every point to itself except for x_{n-1}, which is mapped to $1 - 1/n^2$, is isotonic, and it is not difficult to see that the best approximating similarity transformation satisfies $\|S_n - \varphi_n\|_{\infty(\{x_1,\dots,x_n\})} \in \Omega(\mathrm{d_H}(K, \{x_1, \dots, x_n\}))$. In his paper, Arias-Castro also proves (2.2) to hold (providing convergence rates as well) for functions φ_n that are isotonic on $\{x_1, \dots, x_n\} \cap U_{r_n}(x_i)$ for every $i = 1, \dots, n$ and additionally satisfy, for all $1 \leq i, j, k, l \leq n$,

$$\|x_i - x_j\| < r_n \leq \|x_k - x_l\| \Rightarrow \|\varphi(x_i) - \varphi(x_j)\| \leq \|\varphi(x_k) - \varphi(x_l)\|,$$

assuming that $r_n > 0$ is chosen reasonably. According to Arias-Castro this corresponds to ordinal embeddings of directed, but unweighted k-nearest neighbor graphs that additionally preserve all ordinal relationships $\|A-B\| < \|C-D\|$ among a vertex's k-nearest neighbors. In this context we want to mention that the asymptotic uniqueness of ordinal embeddings of directed, but unweighted k-nearest neighbor graphs (without any additional ordinal relationships) has been outlined in Terada and von Luxburg (2014) as well. Finally, Arias-Castro shows (2.2) to hold in a so-called landmark design. This means that there is a subset $L \subseteq \{x_n : n \in \mathbb{N}\}$ of landmark points, which is assumed to be dense in K, and the functions φ_n are only assumed to be weakly isotonic on $\{x_1, \dots, x_n\} \cap L$ and to preserve ordinal relationships of the form $\|x_i - x_j\| < \|x_i - x_k\|$ for $1 \leq i, j, k \leq n$ and $x_j, x_k \in L$. Assuming the functions φ_n to be isotonic on $\{x_1, \dots, x_n\} \cap L$, Arias-Castro even proves that $\|S_n - \varphi_n\|_{\infty(\{x_1,\dots,x_n\})} \in \mathcal{O}(\mathrm{d_H}(K, \{x_1, \dots, x_n\} \cap L))$.

2.6 Additional lemmas, some proof details, and detailed versions of Lemma 2.8 and Lemma 2.9

In this section we collect some additional lemmas, the details of the proofs of Lemma 2.6 and Proposition 2.12, and the detailed versions of Lemma 2.8 and Lemma 2.9.

2.6.1 Additional lemmas

Lemma 2.14 (Extending a similarity). *Let $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ and $f : \Omega \to \mathbb{R}^d$ be a similarity. Then there exists an affine and surjective similarity $F : \mathbb{R}^d \to \mathbb{R}^d$ (i.e., F is a similarity transformation) such that $F(x) = f(x)$, $x \in \Omega$. The function F is uniquely determined by f if and only if $\mathcal{H}(\Omega) = \mathbb{R}^d$.*

Proof Let $\lambda > 0$ such that $\|f(x) - f(y)\| = \lambda\|x - y\|$, $x, y \in \Omega$. We may assume that $\lambda = 1$ since otherwise we can set $\tilde{f} = (1/\lambda)f$ and $F = \lambda\tilde{F}$ if $\tilde{F}$ is an extension of $\tilde{f}$. In the following we distinguish three cases:

- $0 \in \Omega$ and $f(0) = 0$

  This implies that $\|f(x)\| = \|x\|$, $x \in \Omega$, and because of

$$\begin{aligned}\|f(x)\|^2 - 2\langle f(x), f(x')\rangle + \|f(x')\|^2 &= \|f(x) - f(x')\|^2 \\ &= \|x - x'\|^2 = \|x\|^2 - 2\langle x, x'\rangle + \|x'\|^2\end{aligned}$$

  we can conclude that $\langle f(x), f(x')\rangle = \langle x, x'\rangle$, $x, x' \in \Omega$.

  Let $x_1, \dots, x_n \in \Omega$ form a basis of $[\Omega]$. If $x \in \Omega$ and $x = \sum_{i=1}^n c_i x_i$, then

$$\begin{aligned}\left\| f(x) - \sum_{i=1}^n c_i f(x_i) \right\|^2 &= \|f(x)\|^2 - 2\left\langle f(x), \sum_{i=1}^n c_i f(x_i) \right\rangle + \left\| \sum_{i=1}^n c_i f(x_i) \right\|^2 \\ &= \|x\|^2 - 2\sum_{i=1}^n c_i \langle x, x_i\rangle + \sum_{i=1}^n \sum_{j=1}^n c_i c_j \langle x_i, x_j\rangle \\ &= \sum_{i=1}^n \sum_{j=1}^n c_i c_j \langle x_i, x_j\rangle - 2\sum_{i=1}^n \sum_{j=1}^n c_i c_j \langle x_i, x_j\rangle + \sum_{i=1}^n \sum_{j=1}^n c_i c_j \langle x_i, x_j\rangle \\ &= 0,\end{aligned}$$

  hence $f(x) = \sum_{i=1}^n c_i f(x_i)$. Thus, by setting

$$f'(\tilde{x}) = \sum_{i=1}^n \tilde{c}_i f(x_i) \quad \text{for} \quad \tilde{x} = \sum_{i=1}^n \tilde{c}_i x_i \in [\Omega]$$

  we can define a linear map f' from $[\Omega]$ to $\mathbb{R}^d$ which coincides with f on Ω. Obviously, f' is a linear isometry from $[\Omega]$ onto $f'([\Omega])$. If $[\Omega] \neq \mathbb{R}^d$, we can choose an orthonormal basis of $[\Omega]^\perp$ and one of $f'([\Omega])^\perp$. These comprise the same number of basis vectors since $[\Omega]$ and $f'([\Omega])$ have the same dimension. Let f'' be a linear mapping from $[\Omega]^\perp$ to $f'([\Omega])^\perp$ that maps the orthonormal basis of $[\Omega]^\perp$ onto the one of $f'([\Omega])^\perp$. Then f'' is a linear isometry from $[\Omega]^\perp$ onto $f'([\Omega])^\perp$ and $F = f' \oplus f''$ is a linear isometry from $\mathbb{R}^d$ onto $\mathbb{R}^d$ that is an extension of f.

  Regarding uniqueness: Clearly, if $[\Omega] \neq \mathbb{R}^d$, we can choose different orthonormal bases of $[\Omega]^\perp$ and $f'([\Omega])^\perp$, respectively, or different mappings between them. On the other hand, if $[\Omega] = \mathbb{R}^d$, any linear extension of f to $\mathbb{R}^d$ is uniquely determined by $f(x_1), \dots, f(x_n)$. Since $0 \in \Omega$, we have $\mathcal{H}(\Omega) = [\Omega]$, and because of $f(0) = 0$, any affine extension of f is linear.

- $0 \in \Omega$, but $f(0) \neq 0$

  Define $f' : \Omega \to \mathbb{R}^d$ by $f'(x) = f(x) - f(0)$, $x \in \Omega$. We can apply the previous case to f' and obtain a linear and isometric extension F' of f'. Setting $F = F' + f(0)$ gives the desired extension of f. Obviously, F is uniquely determined if and only if F' is uniquely determined. As we have seen, this is the case if and only if $\mathcal{H}(\Omega) = \mathbb{R}^d$.

- $0 \notin \Omega$ (in fact, one could deal with the second case in the same way as with this case, and so one could merge them into one case "$0 \notin \Omega$, or $0 \in \Omega$, but $f(0) \neq 0$")

Let $x' \in \Omega$ be fixed. Set $\Omega' = \Omega - x'$ and define $f' : \Omega' \to \mathbb{R}^d$ by $f'(x - x') = f(x) - f(x')$, $x \in \Omega$. Then it holds that $0 \in \Omega'$, $f'(0) = 0$, and f' is isometric on Ω'. Let $F' : \mathbb{R}^d \to \mathbb{R}^d$ be the linear and isometric extension of f' according to the first case. Define $F : \mathbb{R}^d \to \mathbb{R}^d$ by $F(x) = F'(x) - F'(x') + f(x')$. Since

$$\begin{aligned} F(x) = F'(x) - F'(x') + f(x') = F'(x - x') + f(x') &= f'(x - x') + f(x') \\ &= f(x) - f(x') + f(x') = f(x) \end{aligned}$$

for $x \in \Omega$, this yields an affine, surjective, and isometric extension of f to $\mathbb{R}^d$. In order to prove the assertion concerning uniqueness of F it suffices to note that $\mathcal{H}(\Omega) = \mathbb{R}^d$ if and only if $[\Omega'] = \mathbb{R}^d$ and that F is unique if and only if F' is unique.

■

Lemma 2.15 (Locally a similarity implies globally a similarity)**.** *Let $\emptyset \neq \Omega \subseteq \mathbb{R}^d$ be open and connected and $f : \Omega \to \mathbb{R}^d$ be locally a similarity. Then f is globally a similarity.*

Proof For $z \in \Omega$ we can choose $\varepsilon_z > 0$ and $\lambda_z > 0$ such that $U_{\varepsilon_z}(z) \subseteq \Omega$ and

$$\|f(u) - f(v)\| = \lambda_z \|u - v\| \quad \forall u, v \in U_{\varepsilon_z}(z)$$

since Ω is open and f is locally a similarity. Fix an arbitrary element $x_0 \in \Omega$ and consider the mapping $f|_{U_{\varepsilon_{x_0}}(x_0)} : U_{\varepsilon_{x_0}}(x_0) \to \mathbb{R}^d$, which is a similarity. According to Lemma 2.14 there exists a unique extension $F_{x_0} : \mathbb{R}^d \to \mathbb{R}^d$, which is a similarity. We show that $f = F_{x_0}|_\Omega$.

Let $y \neq x_0$ be an arbitrary element of Ω. Since any open and connected subset of $\mathbb{R}^d$ is path-connected (e.g., Sutherland, 1975, Proposition 6.4.2), there exists a continuous path $\varphi : [0, 1] \to \Omega$ with $\varphi(0) = x_0$ and $\varphi(1) = y$. Its image $\varphi([0, 1])$ is compact. Hence, we can choose $x_1, \dots, x_n \in \varphi([0, 1])$ with $x_n = y$ such that $\varphi([0, 1])$ is covered by the open balls $U_{\varepsilon_{x_i}}(x_i)$, $i = 0, \dots, n$. W.l.o.g. we may assume that

$$\forall i = 1, \dots, n \ \exists w_i \in \varphi([0, 1]) \subseteq \Omega : w_i \in U_{\varepsilon_{x_{i-1}}}(x_{i-1}) \cap U_{\varepsilon_{x_i}}(x_i).$$

We prove by induction that $f|_{U_{\varepsilon_{x_i}}(x_i)} = F_{x_0}|_{U_{\varepsilon_{x_i}}(x_i)}$ for $i = 0, \dots, n$. This implies $f(y) = F_{x_0}(y)$, and since $y \in \Omega$ was chosen arbitrarily, we can conclude that $f = F_{x_0}|_\Omega$.

The basis ($i = 0$) is clear by construction of F_{x_0}. For the inductive step from $i - 1$ to i let $\varepsilon > 0$ such that $U_\varepsilon(w_i) \subseteq U_{\varepsilon_{x_{i-1}}}(x_{i-1}) \cap U_{\varepsilon_{x_i}}(x_i)$. Note that it immediately follows that $\lambda_{x_i} = \lambda_{x_0}$. According to Lemma 2.14 there exists a unique extension of $f|_{U_\varepsilon(w_i)}$ to a similarity defined on $\mathbb{R}^d$ (which is obviously given by F_{x_0}). There also exists a unique extension of $f|_{U_{\varepsilon_{x_i}}(x_i)}$. However, these extensions have to coincide, and hence we have $f|_{U_{\varepsilon_{x_i}}(x_i)} = F_{x_0}|_{U_{\varepsilon_{x_i}}(x_i)}$. ■

Lemma 2.16 (A diagonal argument)**.** *Let X be an arbitrary non-empty set and $(A_n)_{n \in \mathbb{N}}$, $A_n \subseteq X$, be a sequence of non-empty subsets of X. Let $(\varphi_n)_{n \in \mathbb{N}}$ be a sequence of functions*

$\varphi_n : A_n \to \mathbb{R}^d$. Assume that for every $\varepsilon > 0$ there exists $N(\varepsilon) \in \mathbb{N}$ such that for all $l \geq N(\varepsilon)$ there is a function $S(l,\varepsilon) : X \to \mathbb{R}^d$ with

$$\|\varphi_l - S(l,\varepsilon)\|_{\infty(A_l)} < \varepsilon.$$

Then there exists a sequence of functions $(S_n)_{n\in\mathbb{N}}$, $S_n : X \to \mathbb{R}^d$, with

$$\|\varphi_n - S_n\|_{\infty(A_n)} \to 0 \quad \textit{as } n \to \infty,$$

where every S_n equals a function $S(l_n, \varepsilon_n)$.

Proof We can choose a strictly decreasing sequence of positive reals $(\varepsilon_n)_{n\in\mathbb{N}}$ converging to zero and a strictly increasing sequence of natural numbers $(N_n)_{n\in\mathbb{N}}$ such that for every $l \geq N_n$ there is a function $S(l,\varepsilon_n) : X \to \mathbb{R}^d$ with $\|\varphi_l - S(l,\varepsilon_n)\|_{\infty(A_l)} < \varepsilon_n$. Let $\varepsilon_0 > 0$ and $l_0 \geq N(\varepsilon_0)$ be arbitrary. Set $S_k = S(l_0, \varepsilon_0)$ for $k < N_1$ and $S_k = S(k, \varepsilon_n)$ for $N_n \leq k < N_{n+1}$. In order to show that $\|\varphi_n - S_n\|_{\infty(A_n)} \to 0$, let $\delta > 0$ be arbitrary. Let $n_0 \in \mathbb{N}$ such that $\varepsilon_{n_0} < \delta$. If $m \geq N_{n_0}$, then we have $N_{\tilde{n}} \leq m < N_{\tilde{n}+1}$ for some $\tilde{n} \geq n_0$, and it holds that

$$\|\varphi_m - S_m\|_{\infty(A_m)} = \|\varphi_m - S(m, \varepsilon_{\tilde{n}})\|_{\infty(A_m)} < \varepsilon_{\tilde{n}} \leq \varepsilon_{n_0} < \delta.$$

■

2.6.2 Proof details

Proof of Lemma 2.6: We want to prove Lemma 2.6 in the following slightly more general form:

Let $N \in \mathbb{N}$ and $(\varepsilon_k)_{1\leq k\leq N}$, $(\delta_k)_{1\leq k\leq N}$ be finite sequences of positive real numbers satisfying (for all $1 \leq k \leq N$ such that an expression makes sense)

$$\varepsilon_k < \varepsilon_{k+1}, \quad \delta_k \geq \delta_{k+1}, \quad \varepsilon_k > \varepsilon_j + \delta_j,\ j < k, \quad \varepsilon_N + \delta_N + \max_{j=1,\dots,N-1}(\varepsilon_j + \delta_j) < \frac{1}{2^N}. \tag{2.21}$$

For $k \in \{1,\dots,N\}$ and $i \in \{1,3,\dots,2^k-1\}$ set $x_{k,i} = i/2^k$ and let $y^l_{k,i}$, $y^r_{k,i}$ be arbitrary elements of $(x_{k,i} - \varepsilon_k - \delta_k, x_{k,i} - \varepsilon_k)$ and $(x_{k,i} + \varepsilon_k, x_{k,i} + \varepsilon_k + \delta_k)$, respectively.

Let $\varphi : \{0,1\} \cup \{y^m_{k,i} : m \in \{l,r\}, k \leq N, i \in \{1,3,\dots,2^k-1\}\} \to [0,1]$ be a weakly isotonic function with $\varphi(0) = 0$ and $\varphi(1) = 1$. Then it holds for $k \leq N$ and $i \in \{1,3,\dots,2^k-1\}$ that

$$\varphi(y^l_{k,i}) \in \left(\frac{2^{N-k}i - 1}{2^N}, \frac{2^{N-k}i}{2^N}\right), \quad \varphi(y^r_{k,i}) \in \left(\frac{2^{N-k}i}{2^N}, \frac{2^{N-k}i + 1}{2^N}\right), \tag{2.22}$$

and hence

$$\left|y^m_{k,i} - \varphi(y^m_{k,i})\right| < \frac{1}{2^N}, \quad m \in \{l,r\}, k \leq N, i \in \{1,3,\dots,2^k-1\}. \tag{2.23}$$

Due to (2.21) we have

$$y^l_{k,i} \in \left(\frac{2^{N-k}i-1}{2^N}, \frac{2^{N-k}i}{2^N}\right), \quad y^r_{k,i} \in \left(\frac{2^{N-k}i}{2^N}, \frac{2^{N-k}i+1}{2^N}\right),$$

and thus (2.23) follows from (2.22). We prove (2.22) by induction over N. Note that φ is strictly increasing due to $\varphi(0) = 0$ and $\varphi(1) = 1$ according to Lemma 2.5.

For the basis let $N = 1$. Then we have $y^l_{1,1} \in (0, 1/2)$ and $y^r_{1,1} \in (1/2, 1)$ implying that $|0 - y^l_{1,1}| < |1 - y^l_{1,1}|$ and $|0 - y^r_{1,1}| > |1 - y^r_{1,1}|$. Since φ is weakly isotonic, $\varphi(0) = 0$, and $\varphi(1) = 1$, it follows that $|0 - \varphi(y^l_{1,1})| < |1 - \varphi(y^l_{1,1})|$ and $|0 - \varphi(y^r_{1,1})| > |1 - \varphi(y^r_{1,1})|$ and hence $\varphi(y^l_{1,1}) \in (0, 1/2)$ and $\varphi(y^r_{1,1}) \in (1/2, 1)$.

Assume that the statement holds for N and we want to infer that it also holds for $N + 1$. If the assumptions of the lemma are satisfied for $N + 1$, $(\varepsilon_k)_{1\le k\le N}$, $(\delta_k)_{1\le k\le N}$ and $\varphi|_{\{0,1\}\cup\{y^m_{k,i}: m\in\{l,r\}, k\le N, i\in\{1,3,\ldots,2^k-1\}\}}$ satisfy the assumptions with N, and hence the induction hypothesis yields for $k \le N$ and $i \in \{1, 3, \ldots, 2^k - 1\}$

$$\varphi(y^l_{k,i}) \in \left(\frac{2^{N-k}i-1}{2^N}, \frac{2^{N-k}i}{2^N}\right), \quad \varphi(y^r_{k,i}) \in \left(\frac{2^{N-k}i}{2^N}, \frac{2^{N-k}i+1}{2^N}\right).$$

First, consider $y^l_{N+1,1}$ and $y^r_{N+1,1}$. Due to (2.21) we have

$$\frac{1}{2^{N+1}} + \varepsilon_{N+1} + \delta_{N+1} < \frac{1}{2^N} - \varepsilon_N - \delta_N$$

and hence

$$0 < y^l_{N+1,1} < y^r_{N+1,1} < y^l_{N,1} < y^r_{N,1}.$$

We have

$$|y^l_{N+1,1} - 0| < \frac{1}{2^{N+1}} - \varepsilon_{N+1}$$

and

$$|y^l_{N,1} - y^l_{N+1,1}| > \frac{1}{2^N} - \varepsilon_N - \delta_N - \left(\frac{1}{2^{N+1}} - \varepsilon_{N+1}\right) = \frac{1}{2^{N+1}} + \varepsilon_{N+1} - \varepsilon_N - \delta_N.$$

Because of $\varepsilon_N + \delta_N < 2\varepsilon_{N+1}$ according to (2.21), this yields

$$|y^l_{N+1,1} - 0| < |y^l_{N,1} - y^l_{N+1,1}|,$$

which implies that

$$|\varphi(y^l_{N+1,1}) - 0| < |\varphi(y^l_{N,1}) - \varphi(y^l_{N+1,1})|$$

and

$$2\varphi(y^l_{N+1,1}) < \varphi(y^l_{N,1}),$$

respectively. Due to the induction hypothesis we finally obtain

$$\varphi(y^l_{N+1,1}) < \frac{1}{2}\varphi(y^l_{N,1}) < \frac{1}{2}\frac{1}{2^N} = \frac{1}{2^{N+1}}$$

and hence

$$\varphi(y^l_{N+1,1}) \in \left(0, \frac{1}{2^{N+1}}\right).$$

We have

$$|y^r_{N+1,1} - 0| > \frac{1}{2^{N+1}} + \varepsilon_{N+1}$$

and

$$|y^r_{N,1} - y^r_{N+1,1}| < \frac{1}{2^N} + \varepsilon_N + \delta_N - \left(\frac{1}{2^{N+1}} + \varepsilon_{N+1}\right) = \frac{1}{2^{N+1}} + \varepsilon_N - \varepsilon_{N+1} + \delta_N$$

implying that (due to (2.21))

$$|y^r_{N,1} - y^r_{N+1,1}| < |y^r_{N+1,1} - 0|.$$

It follows that

$$\frac{1}{2^{N+1}} < \frac{1}{2}\varphi(y^r_{N,1}) < \varphi(y^r_{N+1,1}).$$

Because of

$$y^r_{N+1,1} < y^l_{N,1},$$

we have

$$\varphi(y^r_{N+1,1}) < \varphi(y^l_{N,1}) < \frac{1}{2^N} = \frac{2}{2^{N+1}}$$

and hence

$$\varphi(y^r_{N+1,1}) \in \left(\frac{1}{2^{N+1}}, \frac{2}{2^{N+1}}\right).$$

We also obtain

$$\varphi(y^l_{N,1}) \in \left(\frac{1}{2^{N+1}}, \frac{2}{2^{N+1}}\right).$$

In the same manner one can show (2.22) for $y^l_{N+1,2^{N+1}-1}$, $y^r_{N+1,2^{N+1}-1}$ and $y^r_{N,2^N-1}$.

Now, let $i \in \{3,5,\ldots,2^{N+1}-3\}$ be arbitrary. Consider the reduced fractions $\frac{i-1}{2^{N+1}} = \frac{j_1}{2^{k_1}}$ and $\frac{i+1}{2^{N+1}} = \frac{j_2}{2^{k_2}}$ with $1 \leq k_1, k_2 \leq N$ and $j_1 \in \{1,3,\ldots,2^{k_1}-1\}$, $j_2 \in \{1,3,\ldots,2^{k_2}-1\}$. Due to (2.21) we have

$$y^l_{k_1,j_1} < y^r_{k_1,j_1} < y^l_{N+1,i} < y^r_{N+1,i} < y^l_{k_2,j_2} < y^r_{k_2,j_2}.$$

We have to show (2.22) for $y^r_{k_1,j_1}$, $y^l_{N+1,i}$, $y^r_{N+1,i}$ and $y^l_{k_2,j_2}$. We have

$$|y^l_{k_1,j_1} - y^l_{N+1,i}| < \frac{i}{2^{N+1}} - \varepsilon_{N+1} - \left(\frac{i-1}{2^{N+1}} - \varepsilon_{k_1} - \delta_{k_1}\right) = \frac{1}{2^{N+1}} - \varepsilon_{N+1} + \varepsilon_{k_1} + \delta_{k_1}$$

and

$$|y^l_{k_2,j_2} - y^l_{N+1,i}| > \frac{i+1}{2^{N+1}} - \varepsilon_{k_2} - \delta_{k_2} - \left(\frac{i}{2^{N+1}} - \varepsilon_{N+1}\right) = \frac{1}{2^{N+1}} - \varepsilon_{k_2} + \varepsilon_{N+1} - \delta_{k_2}.$$

Since $\delta_{k_1} + \delta_{k_2} + \varepsilon_{k_1} + \varepsilon_{k_2} < 2\varepsilon_{N+1}$ according to (2.21), this yields

$$|y^l_{k_1,j_1} - y^l_{N+1,i}| < |y^l_{k_2,j_2} - y^l_{N+1,i}|$$

and hence

$$|\varphi(y^l_{k_1,j_1}) - \varphi(y^l_{N+1,i})| < |\varphi(y^l_{k_2,j_2}) - \varphi(y^l_{N+1,i})|.$$

Using the induction hypothesis we can conclude that

$$\varphi(y^l_{N+1,i}) < \frac{\varphi(y^l_{k_2,j_2}) + \varphi(y^l_{k_1,j_1})}{2} < \frac{1}{2}\left(\frac{i+1}{2^{N+1}} + \frac{i-1}{2^{N+1}}\right) = \frac{i}{2^{N+1}}.$$

The induction hypothesis also yields

$$\varphi(y^r_{k_1,j_1}) > \frac{i-1}{2^{N+1}},$$

and hence we have

$$\varphi(y^r_{k_1,j_1}) \in \left(\frac{i-1}{2^{N+1}}, \frac{i}{2^{N+1}}\right), \quad \varphi(y^l_{N+1,i}) \in \left(\frac{i-1}{2^{N+1}}, \frac{i}{2^{N+1}}\right).$$

We have

$$|y^r_{k_1,j_1} - y^r_{N+1,i}| > \frac{i}{2^{N+1}} + \varepsilon_{N+1} - \left(\frac{i-1}{2^{N+1}} + \varepsilon_{k_1} + \delta_{k_1}\right) = \frac{1}{2^{N+1}} + \varepsilon_{N+1} - \varepsilon_{k_1} - \delta_{k_1},$$

$$|y^r_{k_2,j_2} - y^r_{N+1,i}| < \frac{i+1}{2^{N+1}} + \varepsilon_{k_2} + \delta_{k_2} - \left(\frac{i}{2^{N+1}} + \varepsilon_{N+1}\right) = \frac{1}{2^{N+1}} - \varepsilon_{N+1} + \varepsilon_{k_2} + \delta_{k_2}$$

and hence (due to (2.21))

$$|y^r_{k_2,j_2} - y^r_{N+1,i}| < |y^r_{k_1,j_1} - y^r_{N+1,i}|.$$

In the same manner as above we can conclude that

$$\varphi(y^r_{N+1,i}) \in \left(\frac{i}{2^{N+1}}, \frac{i+1}{2^{N+1}}\right), \quad \varphi(y^l_{k_2,j_2}) \in \left(\frac{i}{2^{N+1}}, \frac{i+1}{2^{N+1}}\right).$$

■

Remark 2.17.

- *The assumptions* (2.21) *on the sequences* $(\varepsilon_k)_{1\le k\le N}$ *and* $(\delta_k)_{1\le k\le N}$ *are equivalent to* $\varepsilon_k > 0$*,* $\delta_k > 0$*,* $\delta_k \ge \delta_{k+1}$*,* $\varepsilon_k + \delta_k < \varepsilon_{k+1}$*, and* $\varepsilon_N + \delta_N + \varepsilon_{N-1} + \delta_{N-1} < 1/2^N$*.*

- *Sequences* $(\varepsilon_k)_{1\le k\le N}$ *and* $(\delta_k)_{1\le k\le N}$ *satisfying these assumptions always exist. For example, we can choose* $\varepsilon_k = \varepsilon_1 2^{k-1}$ *and* $\delta_k = \varepsilon_1/2$ *with* $\varepsilon_1 < 1/2^{2N+1}$ *as in Section 2.3.*

Proof of Proposition 2.12: Here we want to prove the statement for the case that $\Omega = B_r(z)$ is a closed and bounded ball (for some arbitrary $r > 0$, $z \in \mathbb{R}^d$) and $f : \Omega \to \mathbb{R}^d$ is globally strongly isotonic. How to derive the general result from this special case is shown in Section 2.4.

Consider the set $f(\Omega)$. Since f is continuous according to Lemma 2.10 and Ω is compact, so is $f(\Omega)$. In particular, $f(\Omega)$ is bounded, that is $\operatorname{diam} f(\Omega) < \infty$. We can define a function $\mu : [0, \operatorname{diam}\Omega] \to [0, \operatorname{diam} f(\Omega)]$ as follows:

$$\forall x, y \in \Omega : \|f(x) - f(y)\| = \mu(\|x - y\|).$$

Since f is strongly isotonic, μ is indeed well-defined. Note that μ is defined on the whole interval $[0, \operatorname{diam}\Omega]$ since Ω naturally contains a line segment of length $\operatorname{diam}\Omega$. In order to show that f is a similarity, we have to show that μ is linear, that is $\mu(t) = \lambda t$, $t \in [0, \operatorname{diam}\Omega]$, for some $\lambda > 0$.

It follows from f being strongly isotonic that μ is strictly increasing. Obviously, we have $\mu(0) = 0$. Due to the compactness of Ω and $f(\Omega)$ and f being strongly isotonic, we can conclude that $\mu(\operatorname{diam}\Omega) = \operatorname{diam} f(\Omega)$.

Choose points x_0 and y_0 on the boundary of Ω with $\|x_0 - y_0\| = \operatorname{diam}\Omega$ (consequently, x_0 and y_0 are elements of a straight line going through z). We can write $\mu(t)$ as

$$\mu(t) = \left\| f(x_0) - f\left(x_0 + t\frac{y_0 - x_0}{\|y_0 - x_0\|}\right)\right\|, \quad t \in [0, \operatorname{diam}\Omega].$$

This shows that μ is continuous.

Let $m = (x_0 + y_0)/2$ be the midpoint of the line segment between x_0 and y_0 (in fact, $m = z$). We want to show that $f(m) = (f(x_0) + f(y_0))/2$. If $d = 1$, this immediately follows from f being strongly isotonic. If $d \ge 2$, set $r_0 = x_0 - y_0$ and let $R = [r_0]$ be the linear hull of r_0. Let $\{e_1, \dots, e_{d-1}\}$ be an orthonormal basis of $R^\perp$. We can choose $\varepsilon > 0$ such that all points $p_i^+ = m + \varepsilon e_i$ and $p_i^- = m - \varepsilon e_i$, $i = 1, \dots, d-1$, are elements of Ω (in fact, we can choose any $\varepsilon \le r$). Set $p_0^+ = x_0$ and $p_0^- = y_0$.
Now we have

$$\|m - p_i^+\| = \|m - p_i^-\|, \quad i = 0, \dots, d-1,$$

and

$$\|p_j^+ - p_i^+\| = \|p_j^+ - p_i^-\|, \quad i \ne j \in \{0, \dots, d-1\},$$
$$\|p_j^- - p_i^+\| = \|p_j^- - p_i^-\|, \quad i \ne j \in \{0, \dots, d-1\}.$$

Since f is strongly isotonic, it follows that

$$\|f(m) - f(p_i^+)\| = \|f(m) - f(p_i^-)\|, \quad i = 0, \dots, d-1,$$

and

$$\begin{aligned} \|f(p_j^+) - f(p_i^+)\| &= \|f(p_j^+) - f(p_i^-)\|, \quad i \neq j \in \{0, \dots, d-1\}, \\ \|f(p_j^-) - f(p_i^+)\| &= \|f(p_j^-) - f(p_i^-)\|, \quad i \neq j \in \{0, \dots, d-1\}. \end{aligned}$$

This implies that

$$\big\langle f(p_i^+) - f(p_i^-), f(m) \big\rangle = \left\langle f(p_i^+) - f(p_i^-), \frac{f(p_i^+) + f(p_i^-)}{2} \right\rangle, \quad i = 0, \dots, d-1, \tag{2.24}$$

and

$$\begin{aligned} \left\langle f(p_i^+) - f(p_i^-), f(p_j^+) \right\rangle &= \left\langle f(p_i^+) - f(p_i^-), \frac{f(p_i^+) + f(p_i^-)}{2} \right\rangle, \quad i \neq j \in \{0, \dots, d-1\}, \\ \left\langle f(p_i^+) - f(p_i^-), f(p_j^-) \right\rangle &= \left\langle f(p_i^+) - f(p_i^-), \frac{f(p_i^+) + f(p_i^-)}{2} \right\rangle, \quad i \neq j \in \{0, \dots, d-1\}. \end{aligned} \tag{2.25}$$

We show that under the conditions (2.25) the point $f(m) = (f(p_0^+) + f(p_0^-))/2 = (f(x_0) + f(y_0))/2$ is the unique solution to (2.24):

1. $f(m) = (f(p_0^+)+f(p_0^-))/2$ is a solution to (2.24): Set $j = 0$ and let $i \in \{1, \dots, d-1\}$ be arbitrary in (2.25). Add the first line in (2.25) to the second and divide by two. Hence, $f(m) = (f(p_0^+) + f(p_0^-))/2$ is a solution to (2.24) for $i = 1, \dots, d-1$ and obviously also for $i = 0$.

2. There is a unique solution to (2.24): (2.24) is a linear system involving d equations for the d unknown coordinates of $f(m)$. It suffices to show that the vectors $f(p_i^+) - f(p_i^-)$, $i = 0, \dots, d-1$, are linearly independent. Subtracting the two lines of (2.25) yields

   $$\left\langle f(p_i^+) - f(p_i^-), f(p_j^+) - f(p_j^-) \right\rangle = 0, \quad i \neq j \in \{0, \dots, d-1\}.$$

   We see that the vectors $(f(p_i^+) - f(p_i^-))$, $i = 0, \dots, d-1$, even form an orthogonal system.

Hence, we have $f(m) = (f(x_0) + f(y_0))/2$ and can conclude that $\mu\,(\operatorname{diam}\Omega/2) = \operatorname{diam} f(\Omega)/2$.

By repeating this procedure (once starting with $x_0 = x_0$, $y_0 = m$, once starting with $x_0 = m$, $y_0 = y_0$), we see that

$$\mu\left(\frac{1}{4}\operatorname{diam}\Omega\right) = \frac{1}{4}\operatorname{diam} f(\Omega) \quad \text{and} \quad \mu\left(\frac{3}{4}\operatorname{diam}\Omega\right) = \frac{3}{4}\operatorname{diam} f(\Omega)$$

and iteratively obtain

$$\mu\left(\frac{j}{2^i}\operatorname{diam}\Omega\right)=\frac{j}{2^i}\operatorname{diam}f(\Omega),\quad i\in\mathbb{N},j\in\{0,\dots,2^i\}.$$

Note that Ω being a ball allows us to find a proper ε in each iteration step. Since μ is continuous, this shows

$$\mu(t)=t\,\frac{\operatorname{diam}f(\Omega)}{\operatorname{diam}\Omega}.$$

■

2.6.3 Detailed versions of Lemma 2.8 and Lemma 2.9

Lemma 2.8. *Let $d\geq 2$. Let $N\in\mathbb{N}$ such that*

$$\omega=24\left(\frac{\Gamma(\frac{d}{2}+1)}{\pi^{\frac{d}{2}}}\right)^{\frac{1}{d}}\left(\frac{1}{2^N-1}\right)^{\frac{1}{d}}<\frac{1}{2(d-1)}$$

be fixed. Let $r,\tilde r,\alpha,\tilde\alpha\in\mathbb{R}^d$, let $\mu>0$ and let $(\varepsilon_k)_{1\leq k\leq N}$, $(\delta_k)_{1\leq k\leq N}$ be real sequences such that (for all $1\leq k\leq N$ such that an expression makes sense)

$$\begin{array}{ll}
r,\tilde r>0,\quad \alpha,\tilde\alpha>0, & \varepsilon_k>0,\quad \delta_k>0,\\
\alpha<r,\quad \tilde\alpha<\tilde r, & \varepsilon_{k+1}>\varepsilon_k,\quad \delta_{k+1}\leq\delta_k,\\
r_1=\tilde r_1=1, & \delta_1<\mu,\quad \delta_1<\varepsilon_1,\\
r_j\leq 1,\quad \tilde r_j\leq 1,\quad j=2,\dots,d, & \alpha_1+\delta_1+d\mu<\varepsilon_1,\\
\tilde\alpha_1<\omega,\quad \max_{s=1,\dots,d}\tilde\alpha_s<\frac{1}{2}, & 4\varepsilon_N+4\delta_1+d\mu<\frac{1}{2^N},\\
\rho=\max_{j=2,\dots,d}\frac{\tilde\alpha_j(\tilde r_j+3\sqrt{d-1})}{\tilde r_j-\tilde\alpha_j}<\omega, & \varepsilon_{k+1}>\varepsilon_k+2\delta_1+d\mu+\alpha_1,
\end{array}$$
$$4\mu r_j-4\alpha_j\sqrt{1+(d-1)\mu^2}-4r_j\alpha_j-4r_j\delta_1-4\alpha_j\delta_1-\alpha_j^2>0,\quad j=2,\dots,d, \tag{2.26}$$

and such that all the balls $U^l_{k,i}$, $U^r_{k,i}$ and $U^j_{k,i}$, which we define below, lie in the convex hull of the points $X_1^+,X_1^-,\dots,X_d^+,X_d^-$ defined in the next paragraph.

Define the points $m_s^+,m_s^-,\widetilde m_s^+,\widetilde m_s^-\in\mathbb{R}^d$, $s=1,\dots,d$, by

$$\begin{aligned}
m_1^+&=(r_1/0/0/0/\dots), & m_1^-&=(-r_1/0/0/0/\dots),\\
\widetilde m_1^+&=(\tilde r_1/0/0/0/\dots), & \widetilde m_1^-&=(-\tilde r_1/0/0/0/\dots),\\
m_2^+&=(0/r_2/0/0/\dots), & m_2^-&=(0/-r_2/0/0/\dots),\\
\widetilde m_2^+&=(0/\tilde r_2/0/0/\dots), & \widetilde m_2^-&=(0/-\tilde r_2/0/0/\dots),\quad \textit{and so forth.}
\end{aligned}$$

Let $X_s^+,X_s^-\in\mathbb{R}^d$, $s=1,\dots,d$, be arbitrary elements of $U_{\alpha_s}(m_s^+)$ and $U_{\alpha_s}(m_s^-)$, respectively.

For $k\in\{1,\dots,N\}$, $i\in\{1,3,\dots,2^k-1\}$, and $j\in\{2,\dots,d\}$ set

$$x_{k,i}=-1+\frac{i}{2^{k-1}},\qquad o^j_{k,i}=(x_{k,i}/-\mu/\dots/-/\mu/\underbrace{+\mu}_{jth\ \ entry}/-\mu/\dots/-\mu)\in\mathbb{R}^d,$$

$$u^l_{k,i}=(x_{k,i}-\varepsilon_k/-\mu/\dots/-\mu)\in\mathbb{R}^d,\qquad u^r_{k,i}=(x_{k,i}+\varepsilon_k/-\mu/\dots/-\mu)\in\mathbb{R}^d,$$

and define the open balls

$$U^j_{k,i} = U_{\delta_k}(o^j_{k,i}), \quad U^l_{k,i} = U_{\delta_k}(u^l_{k,i}), \quad U^r_{k,i} = U_{\delta_k}(u^r_{k,i}).$$

Let $z^j_{k,i}$ be an arbitrary element of $U^j_{k,i}$ and $y^l_{k,i}$, $y^r_{k,i}$ be arbitrary elements of $U^l_{k,i}$ and $U^r_{k,i}$, respectively.

Let $\varphi : \{X^+_1, X^-_1, \ldots, X^+_d, X^-_d\} \cup \{z^j_{k,i} : k \leq N, i \in \{1, 3, \ldots, 2^k - 1\}, j \in \{2, \ldots, d\}\} \cup \{y^m_{k,i} : m \in \{l, r\}, k \leq N, i \in \{1, 3, \ldots, 2^k - 1\}\} \to \mathbb{R}^d$ be an isotonic function and assume that

$$\varphi(X^+_s) \in U_{\tilde{\alpha}_s}(\widetilde{m}^+_s), \quad \varphi(X^-_s) \in U_{\tilde{\alpha}_s}(\widetilde{m}^-_s), \quad s = 1, \ldots, d. \tag{$\sharp$}$$

Set $\gamma(-1) = \gamma(1) = \tilde{\alpha}_1$ and $\gamma(0) = \tilde{\alpha}_1 + \frac{d-1}{2}(\omega + \rho)$, and define for $k \in \{2, \ldots, N\}$ and $i \in \{1, 3, \ldots, 2^k - 1\}$ the positive expression $\gamma(-1 + i/2^{k-1})$ recursively by

$$\gamma\left(-1 + \frac{i}{2^{k-1}}\right) = \frac{1}{2}\left(\gamma\left(-1 + \frac{i-1}{2^{k-1}}\right) + \gamma\left(-1 + \frac{i+1}{2^{k-1}}\right) + (d-1)(\omega + 2\rho)\right).$$

Let $N^* < N$ such that $N^* \cdot 2^{N^*} < \frac{1}{5(d+1)(\omega+\rho+\tilde{\alpha}_1)}$. Then we have

$$\begin{aligned} \varphi(y^l_{k,i}) &\in (x_{k,i} - \gamma(x_{k,i}) - \omega, x_{k,i} + \gamma(x_{k,i})) \times (-\rho - \omega, \rho)^{d-1}, \\ \varphi(y^r_{k,i}) &\in (x_{k,i} - \gamma(x_{k,i}), x_{k,i} + \gamma(x_{k,i}) + \omega) \times (-\rho - \omega, \rho)^{d-1} \end{aligned}$$

and hence

$$\left\| y^m_{k,i} - \varphi\left(y^m_{k,i}\right)\right\| < \gamma(x_{k,i}) + \omega + (d-1)(\omega + \rho) < 3d\sqrt{\omega}, \quad m \in \{l, r\},$$

for all $1 \leq k \leq N^*$ and $i \in \{1, 3, \ldots, 2^k - 1\}$.

Remark 2.18. *Using a continuity argument, it is straightforward to see that for any $N \in \mathbb{N}$ there exist $r, \tilde{r}, \alpha, \tilde{\alpha} \in \mathbb{R}^d$, a constant $\mu > 0$, and sequences $(\varepsilon_k)_{1 \leq k \leq N}$, $(\delta_k)_{1 \leq k \leq N}$ satisfying (2.26) and having the property that all the balls $U^l_{k,i}$, $U^r_{k,i}$, and $U^j_{k,i}$ lie in the convex hull of $X^+_1, X^-_1, \ldots, X^+_d, X^-_d$ for any choice of these points within the balls $U_{\alpha_s}(m^+_s)$ and $U_{\alpha_s}(m^-_s)$, respectively.*

Lemma 2.9. *Let $d \geq 2$. Let $N' \in \mathbb{N}$ such that*

$$\omega' = 32\left(\frac{\Gamma(\frac{d}{2}+1)}{\pi^{\frac{d}{2}}}\right)^{\frac{1}{d}} \frac{1}{\sqrt[d]{N'}} < 1$$

and all denominators of fractions in

$$A(\omega') = \frac{1}{4}d\left(\frac{20\omega' + 8\frac{10d\omega'}{4d-1}(4d)^{d-1}}{2 - \omega' - \frac{10d\omega'}{4d-1}(4d)^{d-1}}\right)^2 + 2\sqrt{d}\,\frac{20\omega' + 8\frac{10d\omega'}{4d-1}(4d)^{d-1}}{2 - \omega' - \frac{10d\omega'}{4d-1}(4d)^{d-1}} + 5\omega'$$

are larger than one be fixed. Let $r' < 1$ and $\eta, \delta, \varepsilon > 0$ be real numbers such that

$$r' > \frac{1}{2}, \qquad r' > 1 - \sqrt{A(\omega')/2}, \qquad \sqrt{1 + r'^2} > \sqrt{2} - \sqrt{A(\omega')}, \qquad r' + \eta + \varepsilon < 1,$$
$$\sqrt{(d-1)\eta^2 + (r' + \eta)^2} + \varepsilon < 1, \qquad \sqrt{2} + \eta\sqrt{d} + \varepsilon < 2, \qquad \delta < \frac{1}{3N'},$$
$$2(r' + \eta) - 2\varepsilon > \frac{2N' - 1}{N'} + \delta, \qquad \delta + 2\eta\sqrt{d} + 2\varepsilon < \frac{1}{N'},$$

$$\sqrt{(1-\eta)^2 + (-\eta\tilde{v} - mr')^2 + (d-2)\eta^2} + 2\varepsilon < \sqrt{(1+\eta)^2 + (-\eta\tilde{v} - mr')^2 + (d-2)\eta^2} \quad \text{for } \tilde{v}, m \in \{-1, +1\},$$

$$\sqrt{(-1-\eta\tilde{v})^2 + (r' - \eta)^2 + (d-2)\eta^2} + 2\varepsilon < \sqrt{(-1-\eta\tilde{v})^2 + (r' + \eta)^2 + (d-2)\eta^2} \quad \text{for } \tilde{v} \in \{-1, +1\},$$

$$\sqrt{(r' + \eta(\tilde{v} - \tilde{v}'))^2 + (r' - 2\eta)^2 + \sum_{k=1}^{d-2} \eta^2 (v_k - v_k')^2} + 4\varepsilon < \sqrt{(r' + \eta(\tilde{v} - \tilde{v}'))^2 + (r' + 2\eta)^2 + \sum_{k=1}^{d-2} \eta^2 (v_k - v_k')^2}$$
$$\text{for } \tilde{v}, \tilde{v}', v_k, v_k' \in \{-1, +1\} \quad (k = 1, \dots, d-2). \tag{2.27}$$

Define points $A, B \in \mathbb{R}^d$ and $Z_s^-, Z_s^+ \in \mathbb{R}^d$, $s \in \{2, \dots, d\}$, by

$$A = (-1/0/\dots/0), \quad B = (1/0/\dots/0), \quad Z_2^- = \left(0/-r'/0/0/\dots\right),$$
$$Z_2^+ = \left(0/r'/0/0/\dots\right), \quad Z_3^- = \left(0/0/-r'/0/\dots\right), \quad Z_3^+ = \left(0/0/r'/0/\dots\right), \text{ and so forth.}$$

For $s \in \{2, \dots, d\}$ and $v \in \{-1, 1\}^d$ set $E_{s,v}^- = Z_s^- + \eta v$, $E_{s,v}^+ = Z_s^+ + \eta v$ and let $e_{s,v}^-, e_{s,v}^+ \in \mathbb{R}^d$ be arbitrary elements of $U_\varepsilon(E_{s,v}^-)$ and $U_\varepsilon(E_{s,v}^+)$, respectively. For $i \in \{1, \dots, 2N' - 1\}$ let $x_i \in \mathbb{R}^d$ be an arbitrary element of $U_\delta((-1 + \frac{i}{N'}/0/\dots/0))$.

Let $\varphi : \{A, B\} \cup \{e_{s,v}^-, e_{s,v}^+ : s \in \{2, \dots, d\}, v \in \{-1, 1\}^d\} \cup \{x_i : i = 1, \dots, 2N' - 1\} \to \mathbb{R}^d$ be an isotonic function with $\|\varphi(A) - \varphi(B)\| = 2$. Then there exist a constant C depending only on d and an isometry $S : \mathbb{R}^d \to \mathbb{R}^d$ such that

$$\|A - S(\varphi(A))\| \leq C\sqrt{A(\omega')}, \qquad \|B - S(\varphi(B))\| \leq C\sqrt{A(\omega')},$$
$$\|Z_s^m - S(\varphi(e_{s,\underline{v}}^m))\| \leq C\sqrt{A(\omega')}, \quad m \in \{-, +\}, s \in \{2, \dots, d\},$$

where $\underline{v} = (1/1/1/\dots/1)$.

Remark 2.19. *Using a continuity argument, it is straightforward to see that for any $N' \in \mathbb{N}$ there exist real numbers $r' < 1$ and $\eta, \delta, \varepsilon > 0$ satisfying (2.27). Note that these* *assumptions imply that all the balls $U_\varepsilon(E^-_{s,v})$, $U_\varepsilon(E^+_{s,v})$ and $U_\delta((-1+\frac{i}{N'}/0/\dots/0))$ are contained in $U_1(0)$. In fact, we can choose $\eta,\delta,\varepsilon>0$ so small that (2.27) is satisfied and all these balls are contained in the d-dimensional standard cross-polytope. This is the main argument that we require in order to replace the set K in Part 1 of Theorem 2.3 by a cross-polytope or any closed and convex set that is a superset of a cross-polytope and a subset of the smallest ball containing the cross-polytope as remarked in Section 2.2.*

Chapter 3

Dimensionality estimation from the directed, but unweighted k-nearest neighbor graph

In the previous chapter we have proved Shepard's conjecture. It implies that for a Euclidean data set with known intrinsic dimension abundant ordinal data of the type (1.1) asymptotically contains all cardinal distance information up to rescaling. It is natural to wonder whether knowing the intrinsic dimension is really necessary or whether the dimension of the data set can be inferred from the ordinal distance information. In this chapter, we show that estimating the intrinsic dimension from ordinal data is indeed possible and that not even all ordinal constraints $\|A - B\| < \|C - D\|$ are required, but that the directed, but unweighted k-nearest neighbor graph on the data set is sufficient. We provide two estimators, a naive one and a more elaborate one. Both estimators are shown to be statistically consistent when assuming that data points are sampled from a probability space satisfying certain regularity assumptions. However, further theoretical and experimental evidence shows that the elaborate estimator is highly superior and should be preferred in practice.

3.1 Setup and notation for Chapter 3

In a setting of cardinal distance information dimensionality estimation is a well-studied problem. There were many publications already around the time of the development of multidimensional scaling (Shepard and Carroll, 1966, Trunk, 1968, Bennett, 1969, Fukunaga and Olsen, 1971, Chen and Andrews, 1974, Pettis et al., 1979, Grassberger and Procaccia, 1983) and it has gained renewed attention after the invention of manifold learning algorithms like Isomap (Tenenbaum et al., 2000) or Locally Linear Embedding (Roweis and Saul, 2000) (Camastra and Vinciarelli, 2002, Kégl, 2002, Costa and Hero, 2004, Levina and Bickel, 2004, Costa et al., 2005, Hein and Audibert, 2005, Farahmand et al., 2007, Sricharan et al., 2010, Eriksson and Crovella, 2012, Ceruti et al., 2014). A recent survey about these and further methods is provided in Camastra and Staiano (2016). Most of these methods are formulated in the following general setup: Let $\mathcal{X}' \subseteq \mathbb{R}^d$

be a low-dimensional set, f a continuous probability density function on $\mathcal{X}'$, and $\varphi : \mathcal{X}' \to \mathcal{M} \subseteq \mathbb{R}^D$ a smooth embedding of $\mathcal{X}'$ in a high-dimensional space. Points $x'_1, \ldots, x'_n \in \mathcal{X}'$ are drawn from f. They are embedded into the observation space $\mathbb{R}^D$ via φ and possibly disturbed by noise $\eta_i \in \mathbb{R}^D$, resulting in the data set $\mathcal{D} = \{x_1, \ldots, x_n\}$ with $x_i = \varphi(x'_i) + \eta_i$, $i = 1, \ldots, n$. The task is to infer the intrinsic dimension d, where all the existing methods assume to be given coordinates $(x_i^1, \ldots, x_i^D)$ or distance values $\|x_i - x_j\|_{\mathbb{R}^D}$. We consider the same setup, but instead of assuming to observe coordinates or distance values, we assume to be only given the directed, but unweighted k-NN graph on $\mathcal{D}$ for some $k \ll n$ and constructed with respect to $\| \cdot \|_{\mathbb{R}^D}$. We refer to this k-NN graph on $\mathcal{D}$ as G. Recall from Section 1.3 that G encodes knowledge about memberships to the sets of k nearest neighbors of data points. It has the vertex set $V = \{1, \ldots, n\} \simeq \mathcal{D}$ and a directed, unweighted edge from i to j if and only if x_j is among the k nearest sample points to x_i with respect to $\| \cdot \|_{\mathbb{R}^D}$. We denote an edge from i to j by $i \to j$.

Note that although the problem of dimensionality estimation in the described setup is mathematically well-defined, it comes along with an inherent problem in practice (regardless of whether we can observe coordinates, distances, or the k-NN graph on $\mathcal{D}$): The data set $\mathcal{D}$ "looks" different at different scales, on the one hand due to the presence of noise, on the other hand due to the curvature of the manifold $\mathcal{M}$. For example, if the data points lie in a small ε-tube around a one-dimensional sphere in $\mathbb{R}^2$, we will only be able to identify the one-dimensionality of $\mathcal{D}$ if we look at it on a proper scale. If we "zoom in too closely", say we consider an ε-ball of the data set, it will appear to have dimension 2. If we "zoom out very far", then $\mathcal{D}$ will even look like a single point and thus may be considered as zero-dimensional. In our case, the scale at which we look at $\mathcal{D}$ is controlled by the parameter k: the larger k, the more we "zoom out".

In the following, we denote by $B_{\mathrm{SP}}(i, r)$ the closed ball with center $i \in V$ and radius $r > 0$ in the graph G with respect to the directed shortest path distance d_{SP}, that is $B_{\mathrm{SP}}(i, r) = \{j \in V : d_{\mathrm{SP}}(i, j) \leq r\}$. By λ_d we denote the d-dimensional Lebesgue measure and by $\eta_d = \lambda_d(B_1(0))$ the volume of the d-dimensional unit ball. For $\mathcal{X} \subseteq \mathbb{R}^d$ we denote its topological boundary by $\partial(\mathcal{X})$.

3.2 Our estimators

In this section we describe two strategies that yield an estimate of the intrinsic dimension d based on the directed, but unweighted k-NN graph G on the data set $\mathcal{D}$. Both methods have in common that they estimate quantities related to d locally around sample points and then combine these local estimates to one global estimate for d.

3.2.1 Estimator based on doubling property

Recall the doubling property of the Lebesgue measure λ_d: for any $x \in \mathbb{R}^d$ and $r > 0$ we have $\lambda_d(B(x, 2r)) = 2^d \lambda_d(B(x, r))$. Consequently, we can determine the dimension d from the volumes of two balls with radius r and $2r$, respectively, by

$$d = -\log_2(\lambda_d(B(x, r))/\lambda_d(B(x, 2r))).$$

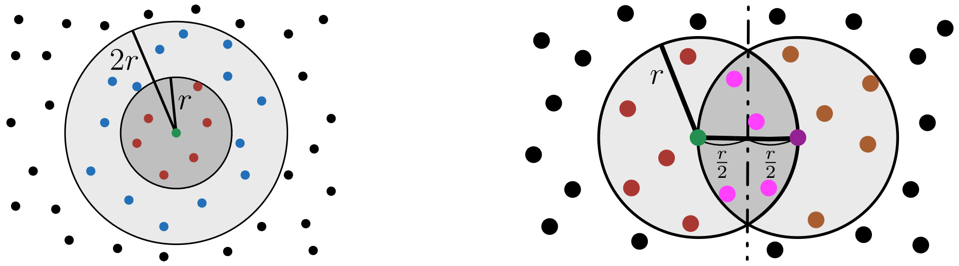


Figure 3.1: The idea behind our estimators E_{DP} and E_{CAP}. Left: E_{DP} is based on the fact that points x_j with $j \in B_{\mathrm{SP}}(i,1)$ (green + red) approximately fill $B(x_i, r)$ and points x_j with $j \in B_{\mathrm{SP}}(i,2)$ (green + red + blue) approximately fill $B(x_i, 2r)$. Right: E_{CAP} makes use of the following observation: if x_{j_0} (in purple) is a point sitting at the boundary of $B(x_i, r)$, points x_j with $j \in B_{\mathrm{SP}}(i,1) \cap B_{\mathrm{SP}}(j_0,1)$ (green + purple + magenta) fill two disjoint spherical caps with height $r/2$ of the balls with radius r.

This is the property that we are exploiting in our first naive estimator. To carry it over to the finite sample setting, fix any sample point x_i that is sufficiently far from the boundary of $\mathcal{M}$ and consider $B_{\mathrm{SP}}(i,1)$ and $B_{\mathrm{SP}}(i,2)$. If the sample size n is large enough and k is relatively small, points x_j with $j \in B_{\mathrm{SP}}(i,2)$ lie in such a small neighborhood of x_i on $\mathcal{M}$ that we can actually think of $\mathcal{M}$ as flat and identify it with $\mathbb{R}^d$. Here we do not take the noise into account. Then, as we will prove in Section 3.3, the balls $B_{\mathrm{SP}}(i,1)$ and $B_{\mathrm{SP}}(i,2)$ in G approximately correspond to balls $B(x_i, r)$ and $B(x_i, 2r)$ in $\mathbb{R}^d$, for some small radius r. This is illustrated on the left side of Figure 3.1: in green we see the point x_i, in red points x_j with $j \in B_{\mathrm{SP}}(i,1)$, and in blue points x_j with $j \in B_{\mathrm{SP}}(i,2) \setminus B_{\mathrm{SP}}(i,1)$. On the small balls $B(x_i, r)$ and $B(x_i, 2r)$ we can consider the density function f as roughly constant and obtain

$$L_{\mathrm{DP}}(i) := \frac{k+1}{|B_{\mathrm{SP}}(i,2)|} = \frac{|B_{\mathrm{SP}}(i,1)|}{|B_{\mathrm{SP}}(i,2)|} \approx \frac{n\ f(x_i)\ \lambda_d(B(x_i,r))}{n\ f(x_i)\ \lambda_d(B(x_i,2r))} = \frac{1}{2^d}.$$

Hence, an estimate of d is given by $-\log_2 L_{\mathrm{DP}}(i)$. However, in order to obtain a more robust estimator we average over $L_{\mathrm{DP}}(i)$ for various vertices $i \in A \subseteq V$. With

$$L_{\mathrm{DP}}(A) := \frac{1}{|A|} \sum_{i \in A} L_{\mathrm{DP}}(i)$$

this leads to our first dimension estimator

$$E_{\mathrm{DP}}(A) := -\log_2 L_{\mathrm{DP}}(A).$$

By construction, E_{DP} resembles classical dimension estimators exploiting the doubling property of the Lebesgue measure (e.g., the method by Grassberger and Procaccia, 1983). However, while all the existing methods explicitly use distance values, we make use of the fact that neighborhood balls of i in G approximately correspond to neighborhood balls of x_i in $\mathbb{R}^d$ (see Section 3.3 for details).

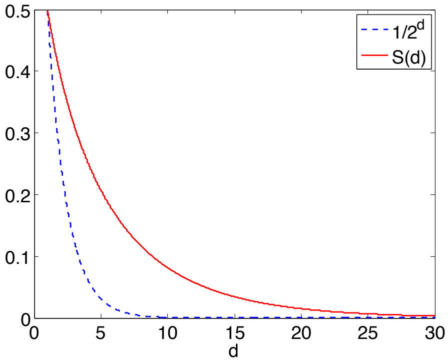


Figure 3.2: The functions $T(d) = 1/2^d$ and S. The latter is more well-behaved in terms of inversion.

3.2.2 Estimator based on spherical caps

Our second estimator relies on a different geometric idea: Fix $x, y \in \mathbb{R}^d$ with $\|x-y\|_{\mathbb{R}^d} = r$ and consider the set $B(x,r) \cap B(y,r)$. This set is the union of two congruent and disjoint (except for their shared base) spherical caps with height $r/2$ of a ball with radius r. An illustration is shown on the right side of Figure 3.1 (dark grey area). According to Li (2011), the volume of such a cap is given by

$$\frac{1}{2}\eta_d r^d I_{\frac{3}{4}}\left(\frac{d+1}{2}, \frac{1}{2}\right),$$

where $I_x(a,b)$ is the regularized incomplete beta function. Consequently,

$$\frac{\lambda_d(B(x,r) \cap B(y,r))}{\lambda_d(B(x,r))} = I_{\frac{3}{4}}\left(\frac{d+1}{2}, \frac{1}{2}\right) =: S(d), \tag{3.1}$$

which is a quantity injectively depending on $d \geq 0$. A plot of the function S can be seen in Figure 3.2. Hence, the dimension d can be retrieved by inverting S. This cannot be done analytically, but can easily be solved numerically.

Our goal is now to follow this idea in the finite sample setting. As in the previous section when deriving our estimator E_{DP}, we fix a sample point x_i and replace $B(x_i, r)$ by $B_{\text{SP}}(i, 1)$. We need to find a vertex j_0 such that x_{j_0} sits on the boundary of $B(x_i, r)$ and then consider $|B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j_0,1)| \approx nf(x_i)\lambda_d(B(x_i,r) \cap B(x_{j_0}, r))$. Because $|B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j,1)|$ tends to decrease as the distance between x_i and x_j increases, we can find such a vertex j_0 as the minimizer of the term $|B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j,1)|$ over vertices j that are connected to i. This leads to

$$L_{\text{CAP}}(i) := \frac{\min_{j \in V : i \to j} |B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j,1)|}{k+1} \approx S(d).$$

An estimate for d is then given by $S^{-1}(L_{\text{CAP}}(i))$. In Section 3.3 we will show that this intuitive derivation is indeed correct. As in the previous section, we make the estimator more robust by averaging over $L_{\text{CAP}}(i)$ for various vertices $i \in A \subseteq V$. With

$$L_{\text{CAP}}(A) := \frac{1}{|A|}\sum_{i \in A} L_{\text{CAP}}(i)$$

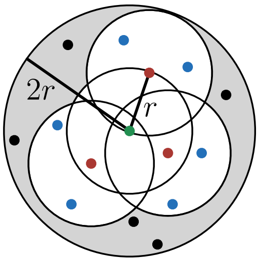


Figure 3.3: Explanation for the bias of E_{DP}: the union of the small balls approximates the large ball, but ignores a substantial part close to its boundary (shaded area).

our second dimension estimator is given by

$$E_{\mathrm{CAP}}(A) := S^{-1}(L_{\mathrm{CAP}}(A)).$$

3.2.3 First comparison of E_{DP} and E_{CAP}

A closer look at the construction of our two estimators E_{DP} and E_{CAP} reveals two reasons why E_{CAP} might perform better than E_{DP}. This theoretical finding will later be confirmed by our experiments in Section 3.4.

The rationale of E_{DP} is to find an expression L_{DP} that approximates $T(d) := 1/2^d$, whereas in E_{CAP} we find an expression L_{CAP} that approximates $S(d)$ as given in (3.1). In both cases, the final estimate is obtained by inverting T or S, respectively, to retrieve d. Inverting a function h is easy and robust in areas where the function is reasonably steep, but is difficult in areas where it is flat. In flat areas of h, small deviations in $h(x)$ can lead to large deviations in $x = h^{-1}(h(x))$. Now consider the plot of the functions T and S in Figure 3.2. It is plain to see that S has a much larger range where it is well-behaved (say, from $d = 1$ to 20) than T (say, from $d = 1$ to 8). Consequently, in the range of $d = 9$ to $d = 20$ the estimator E_{CAP} is still rather robust against deviations of $L_{\mathrm{CAP}}(A)$ from $S(d)$, while small deviations of $L_{\mathrm{DP}}(A)$ from $1/2^d$ lead to large deviations of $E_{\mathrm{DP}}(A) = -\log_2(L_{\mathrm{DP}}(A))$ from d. This is the first observation that suggests an advantage of E_{CAP} over E_{DP}.

Our second insight is that E_{DP} might systematically underestimate the true dimension, in particular if the true dimension is high. The estimator E_{DP} is based on approximating $B(x_i, 2r)$ by $B_{\mathrm{SP}}(i, 2)$. However, as Figure 3.3 shows, there is a bias in this approximation: $B_{\mathrm{SP}}(i, 2)$ is the union of $B_{\mathrm{SP}}(i, 1)$ and balls $B_{\mathrm{SP}}(j, 1)$ for vertices j with $i \to j$. Hence, $B_{\mathrm{SP}}(i, 2)$ actually corresponds to points in the union of $B(x_i, r)$ and balls $B(x_j, r)$. In the limit, as n and k go to infinity, this union approximates $B(x_i, 2r)$ up to arbitrary precision (compare with Section 3.3), but if n and k are not too large, this union is only a poor approximation of $B(x_i, 2r)$, just filling it partially and ignoring a substantial part close to the boundary of $B(x_i, 2r)$. As a consequence, we systematically underestimate $\lambda_d(B(x_i, 2r))$ and thus underestimate d. This effect is increased if d is high because of the fact that in high-dimensional spaces almost all of the volume of a ball is concentrated in a thin shell close to the ball's boundary.

3.2.4 Implementation of our estimators

There is no closed form for the inverse of the function S as given in (3.1). If one is merely interested in an integer estimate of d, the simplest possibility is to set $E_{\text{CAP}}(A)$ to $d^* = \operatorname{argmin}_{d \in \mathbb{N}_0} |S(d) - L_{\text{CAP}}(A)|$. In case one rather wants to have a real-valued estimate, the simplest way is to create a fine-meshed lookup table of argument-value pairs of S, which can be reused every time one wants to apply E_{CAP}.

Assuming that G is given by its unsorted adjacency lists, both $L_{\text{DP}}(i)$ and $L_{\text{CAP}}(i)$ can be implemented with $\mathcal{O}(k^2 \log k)$ time, where the implementation requires $\mathcal{O}(k^2)$ and $\mathcal{O}(k)$, respectively, auxiliary space. This can be done by sorting the union of the adjacency lists of vertex i and of vertices connected to i or sorting each of these adjacency lists in order to compute $|B_{\text{SP}}(i,2)|$ or $\min_{j \in V : i \to j} |B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j,1)|$. Hence, assuming that the inversion of S as addressed in the previous paragraph can be done in constant time and space, the computation of $E_{\text{DP}}(A)$ or $E_{\text{CAP}}(A)$ can be performed in $\mathcal{O}(|A|\; k^2 \log k)$ time and $\mathcal{O}(|A| + k^2)$ or $\mathcal{O}(|A| + k)$ space. If G is given by its adjacency matrix J with $J_{ij} = 1$ if $i \to j$ and 0 otherwise, it is usually faster to make use of the following observations, in particular when $|A|$ is large (e.g., $A = V$):

$$\left(\tilde{J} \cdot \tilde{J}\right)_{ij} > 0 \Leftrightarrow j \in B_{\text{SP}}(i,2), \qquad \left(\tilde{J} \cdot \tilde{J}^T\right)_{ij} = \left|B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j,1)\right|.$$

Here, $\tilde{J}$ is the matrix J with the diagonal entries set to 1. Note that both J and $\tilde{J}$ are sparse matrices with k and $k+1$, respectively, non-zero entries per row.

As we will prove in the next section, both our estimators are statistically consistent for any prespecified choice of $A \subseteq \{1, \ldots, n\}$. However, the variance of $E_{\text{DP}}(A)$ or $E_{\text{CAP}}(A)$ decreases as the size of A increases, and so we suggest to choose $|A|$ as large as one can afford due to computational reasons. Our experiments of Section 3.4 show that the variance of $E_{\text{CAP}}(A)$ decreases even almost like $1/|A|$. This is the rate that one would expect (after a linearization of the function S) if the local statistics $L_{\text{CAP}}(i)$ were independent among $i \in V$. The variance of $E_{\text{DP}}(A)$ decreases more slowly. Apparently, the local statistics $L_{\text{DP}}(i)$ for $i \in V$ are more correlated than the local statistics $L_{\text{CAP}}(i)$. This is not surprising given that $L_{\text{DP}}(i)$ is based on a larger neighborhood of the sample point x_i than $L_{\text{CAP}}(i)$.

3.3 Consistency

In this section we prove that both our estimators $E_{\text{DP}}(A)$ and $E_{\text{CAP}}(A)$, for any prespecified $A \subseteq \{1, \ldots, n\}$, converge in probability to the true dimension d as $n \to \infty$, assuming $k = k(n)$ is chosen reasonably. By prespecified $A \subseteq \{1, \ldots, n\}$ we mean that A is chosen without any information about the data set $\mathcal{D}$ or the graph G. We only consider the case of a flat manifold $\mathcal{M}$ (i.e., φ is a global isometry) and when there is no noise (compare with Section 3.1). This is the relevant case for combining our results of this chapter with those of Chapter 2. It allows us to drop the assumption of known intrinsic dimension in our result that for a Euclidean data set abundant ordinal data asymptotically contains all cardinal distance information up to rescaling. In this case, we simply consider $\mathcal{X} \subseteq \mathbb{R}^d$, a sample $\mathcal{D} = \{x_1, \ldots, x_n\} \subseteq \mathcal{X}$ drawn from a probability density function f on $\mathcal{X}$, and the directed, but unweighted k-NN graph on $\mathcal{D}$ constructed with respect to $\|\cdot\|_{\mathbb{R}^d}$. We make the following regularity assumptions:

Assumptions 3.1 (Regularity assumptions on $\mathcal{X}$ and f)**.**

1. *The domain $\mathcal{X} \subseteq \mathbb{R}^d$ is compact and has boundary of measure zero, that is*
$$\lambda_d(\partial\mathcal{X}) = 0. \tag{3.2}$$

2. *The boundary of the domain $\mathcal{X}$ is regular in the sense that there exist constants $\alpha, \varepsilon_0 > 0$ such that*
$$\lambda_d(B(x,\varepsilon) \cap \mathcal{X}) \geq \alpha \cdot \lambda_d(B(x,\varepsilon)), \quad x \in \mathcal{X}, \varepsilon < \varepsilon_0. \tag{3.3}$$

3. *The density function $f : \mathcal{X} \to \mathbb{R}$ is lower and upper bounded by $f_{min} > 0$ and $f_{max} < \infty$, respectively, that is*
$$0 < f_{min} \leq f(x) \leq f_{max} < \infty, \quad x \in \mathcal{X}. \tag{3.4}$$

4. *The density function f is Lipschitz continuous with constant $L \geq 0$, that is*
$$|f(x) - f(y)| \leq L\|x - y\|, \quad x, y \in \mathcal{X}. \tag{3.5}$$

Under these assumptions we have the following theorem.

Theorem 3.2 (Consistency of E_{DP} and E_{CAP})**.** *Let $\mathcal{D} = \{x_1, \ldots, x_n\} \subseteq \mathcal{X} \subseteq \mathbb{R}^d$ be an i.i.d. sample from the probability density function f on $\mathcal{X}$ and G be the directed, but unweighted k-NN graph on $\mathcal{D}$ with respect to $\|\cdot\|_{\mathbb{R}^d}$. Let the Assumptions 3.1 on $\mathcal{X}$ and f hold. Given G as input, both $E_{DP}(A)$ and $E_{CAP}(A)$, for any prespecified $A \subseteq \{1, \ldots, n\}$, converge in probability to the true dimension d as $n \to \infty$ if $k = k(n)$ satisfies $k \in o(n)$,* $\log n \in o(k)$*, and there exists $k' = k'(n)$ with $k' \in o(k)$ and* $\log n \in o(k')$.

The growth conditions on k are the ones to be expected for random k-NN graphs (compare with von Luxburg et al., 2014, Section 4). There are several ways of choosing k and k' in order to satisfy them. For example, we could choose $k = (\log n)^{1+\tau}$ and $k' = (\log n)^{1+\tau/2}$ for some $\tau > 0$.

By the continuous mapping theorem (e.g., van der Vaart, 1998, Theorem 2.3) it is sufficient to prove convergence in probability of $L_{\mathrm{DP}}(A)$ and $L_{\mathrm{CAP}}(A)$ to $1/2^d$ and $S(d)$, respectively, since $-\log_2$ and S^{-1} are continuous functions on $\mathbb{R}_{>0}$ and $(0, S(0))$, respectively. The convergence of $L_{\mathrm{DP}}(A)$ or $L_{\mathrm{CAP}}(A)$ follows from the convergence of $L_{\mathrm{DP}}(i)$ or $L_{\mathrm{CAP}}(i)$ for every $i \in \{1, \ldots, n\}$. Instead of showing convergence of $L_{\mathrm{DP}}(i)$ we can show convergence of $1/L_{\mathrm{DP}}(i)$ since $x \mapsto 1/x$ is continuous on $\mathbb{R}_{>0}$.

Convergence in probability of random variables U_n to a constant $U \in \mathbb{R}$ can be established by proving that
$$e(\delta, n)U - g(\delta, n) \leq U_n \leq E(\delta, n)U + G(\delta, n) \tag{3.6}$$
holds with probability at least $Pr(\delta, n)$ for all $0 < \delta < \delta_0$ and $n \geq N(\delta)$, where $e(\delta, n), E(\delta, n) \to 1$, $g(\delta, n), G(\delta, n) \to 0$ as $n \to \infty$, $\delta \to 0$, and $Pr(\delta, n) \to 1$ as $n \to \infty$

(for fixed δ). We obtain an inequality of the type (3.6) for $1/L_{\mathrm{DP}}(i)$ and $L_{\mathrm{CAP}}(i)$ in a number of steps, which we formulate as separate propositions and lemmas. A central role will be played by the k-NN radius of $x_i \in \mathcal{D}$ given by

$$r_{k,n}(x_i) = \max\{\|x_i - x_j\| : i \to j \text{ in } G\}.$$

Furthermore, we will repeatedly encounter a quantity $u_{k,n}$ given by

$$u_{k,n} = \left(\frac{k}{(n-1)\eta_d}\right)^{1/d}.$$

Note that the conditions on k imply that $u_{k,n} \to 0$ as $n \to \infty$. All following statements hold for $n \in \mathbb{N}$ sufficiently large and $\delta > 0$ sufficiently small. The constants $c_1, \dots, c_6$ in Propositions 3.3 and 3.5 and Lemmas 3.8 and 3.11 depend on $d, \lambda_d(\mathcal{X}), \alpha, f_{min}$, and L.

We start by showing that for all sample points $x_i \in \mathcal{D}$ sufficiently distant to the boundary of $\mathcal{X}$ the k-NN radius $r_{k,n}(x_i)$ is concentrated.

Proposition 3.3 ($r_{k,n}(x_i)$ is concentrated)**.** *There exist* $c_1, c_2 > 0$ *such that we have*

$$\underline{R}_{k,n}(x_i, \delta) \le r_{k,n}(x_i) \le \overline{R}_{k,n}(x_i, \delta) \tag{3.7}$$

for all $x_i \in \mathcal{D}$ *sufficiently distant to* $\partial\mathcal{X}$ *with probability at least* $1 - 2n\exp(-c_1\delta^2 k)$, *where*

$$\underline{R}_{k,n}(x_i, \delta) = \frac{1}{(1+\delta)(1+c_2(k/(n-1))^{1/d})} \cdot \frac{u_{k,n}}{f(x_i)^{1/d}},$$
$$\overline{R}_{k,n}(x_i, \delta) = \frac{1}{(1-\delta)(1-c_2(k/(n-1))^{1/d})} \cdot \frac{u_{k,n}}{f(x_i)^{1/d}}.$$

Proof This can be shown by standard concentration arguments. For example, our proof is very similar to the one of Part 1 of Proposition 30 in von Luxburg et al. (2014). It uses the Angluin-Valiant-Inequality (Angluin and Valiant, 1979, Proposition 2.4):

> *Let* Z *be a binomially distributed random variable and* $0 \le \beta \le 1$. *Denote by* $\mathrm{E}(Z)$ *the expectation of* Z. *Then we have:*
>
> 1. $Pr(Z \ge (1+\beta)\,\mathrm{E}(Z)) \le \exp\left(-\frac{\beta^2}{3}\,\mathrm{E}(Z)\right)$
> 2. $Pr(Z \le (1-\beta)\,\mathrm{E}(Z)) \le \exp\left(-\frac{\beta^2}{2}\,\mathrm{E}(Z)\right)$

Set $\lambda_{k,n}(x_i) = Lu_{k,n}/(f(x_i)^{1/d} f_{min})$. We fix a sample point x_i with distance larger than

$$a'(x_i) := (1+\delta)^{-1/d}(1+\lambda_{k,n}(x_i))^{-1/d}\frac{u_{k,n}}{f(x_i)^{1/d}}$$

from the boundary of $\mathcal{X}$. Then the closed ball $B = B(x_i, a'(x_i))$ is fully contained in $\mathcal{X}$. For $y \in B$ we have due to (3.5)

$$|f(x_i) - f(y)| \le L\|x_i - y\| \le La'(x_i) \le L\frac{u_{k,n}}{f(x_i)^{1/d}}$$

and hence

$$\begin{aligned} f(y) &\le f(x_i) + L\frac{u_{k,n}}{f(x_i)^{1/d}} \le f(x_i) + \frac{f(x_i)}{f_{min}}\frac{Lu_{k,n}}{f(x_i)^{1/d}} = f(x_i)(1+\lambda_{k,n}(x_i)), \\ f(y) &\ge f(x_i) - L\frac{u_{k,n}}{f(x_i)^{1/d}} \ge f(x_i) - \frac{f(x_i)}{f_{min}}\frac{Lu_{k,n}}{f(x_i)^{1/d}} = f(x_i)(1-\lambda_{k,n}(x_i)). \end{aligned}$$

Let Z_B be a random variable given as the number of points in B if $n-1$ points are drawn i.i.d. from f. Note that Z_B is binomially distributed. We obtain an upper and a lower bound for its expectation $\mathrm{E}(Z_B)$ as follows:

$$\begin{aligned} \mathrm{E}(Z_B) = (n-1)\int_B f(x)dx &\le (n-1)f(x_i)(1+\lambda_{k,n}(x_i))\lambda_d(B) \\ &= (n-1)f(x_i)(1+\lambda_{k,n}(x_i))\eta_d a'(x_i)^d = \frac{k}{1+\delta} \end{aligned}$$

and similarly we obtain

$$\mathrm{E}(Z_B) \ge \frac{k}{1+\delta}\frac{1-\lambda_{k,n}(x_i)}{1+\lambda_{k,n}(x_i)}.$$

We have

$$Pr\left(r_{k,n}(x_i) \le a'(x_i)\right) = Pr(Z_B \ge k) = Pr\left(Z_B \ge k\frac{1+\delta}{1+\delta}\right) \le Pr(Z_B \ge \mathrm{E}(Z_B)(1+\delta)),$$

and it follows from the Angluin-Valiant-Inequality that

$$Pr\left(r_{k,n}(x_i) \le a'(x_i)\right) \le \exp\left(-\frac{\delta^2}{3}\,\mathrm{E}(Z_B)\right) \le \exp\left(-\frac{\delta^2}{3}\frac{k}{1+\delta}\frac{1-\lambda_{k,n}(x_i)}{1+\lambda_{k,n}(x_i)}\right).$$

It follows because of (3.4) that for n sufficiently large and δ sufficiently small we have

$$r_{k,n}(x_i) \le \frac{1}{(1+\delta)\left(1+\frac{L}{\eta_d{}^{1/d}f_{min}{}^{1+1/d}}(k/(n-1))^{1/d}\right)}\cdot\frac{u_{k,n}}{f(x_i)^{1/d}} =: \underline{R}_{k,n}(x_i,\delta)$$

with probability at most $\exp(-\delta^2 k/6)$.

Analogously, we can prove that

$$r_{k,n}(x_i) \ge \frac{1}{(1-\delta)\left(1-\frac{L}{\eta_d{}^{1/d}f_{min}{}^{1+1/d}}(k/(n-1))^{1/d}\right)}\cdot\frac{u_{k,n}}{f(x_i)^{1/d}} =: \overline{R}_{k,n}(x_i,\delta)$$

with probability at most $\exp(-\delta^2 k/6)$ for every x_i with distance larger than

$$a''(x_i) := (1-\delta)^{-1/d}(1-\lambda_{k,n}(x_i))^{-1/d}\frac{u_{k,n}}{f(x_i)^{1/d}}$$

from the boundary of $\mathcal{X}$. Note that we have

$$a'(x_i) \le a''(x_i) \le (1-\delta)^{-1/d}\left(1 - \frac{Lu_{k,n}}{{f_{min}}^{1+1/d}}\right)^{-1/d} \frac{u_{k,n}}{{f_{min}}^{1/d}} =: a. \tag{3.8}$$

Applying a union bound to both events $r_{k,n}(x_i) \le \underline{R}_{k,n}(x_i,\delta)$ and $r_{k,n}(x_i) \ge \overline{R}_{k,n}(x_i,\delta)$ and all x_i with distance larger than a from the boundary of $\mathcal{X}$ yields the statement. ■

Note that $\underline{R}_{k,n}(x_i,\delta), \overline{R}_{k,n}(x_i,\delta) \to 0$ as $n \to \infty$ under the conditions on k. This convergence is uniform over x_i due to (3.4).

The next lemma shows that for nearby sample points x_i and x_j the k-NN radii $r_{k,n}(x_i)$ and $r_{k,n}(x_j)$ will not be too different.

Lemma 3.4 (Locally $r_{k,n}$ varies only slightly). *Assume the event considered in Proposition 3.3 holds. Then we have for a sample point $x_i \in \mathcal{D}$ sufficiently distant to $\partial\mathcal{X}$ and all $y \in \mathcal{D} \cap B(x_i, \overline{R}_{k,n}(x_i,\delta))$*

$$r_{k,n}(y) \ge \underline{R}_{k,n}(x_i,\delta) - a_{k,n}(\delta){u_{k,n}}^{1+1/d}, \tag{3.9}$$

$$r_{k,n}(y) \le \overline{R}_{k,n}(x_i,\delta) + a_{k,n}(\delta){u_{k,n}}^{1+1/d}, \tag{3.10}$$

where $a_{k,n}(\delta) > 0$ converges to a positive constant as $n \to \infty$, $\delta \to 0$, assuming the conditions on k hold.

Proof The lemma follows from the Lipschitz continuity of f. We have

$$\overline{R}_{k,n}(x_i,\delta) = \frac{1}{(1-\delta)(1-c_2(k/(n-1))^{1/d})} \cdot \frac{u_{k,n}}{f(x_i)^{1/d}} \le \frac{2}{{f_{min}}^{1/d}} \cdot u_{k,n}$$

for n sufficiently large and δ sufficiently small due to the conditions on k and (3.4). If x_i has distance larger than $a + 2{f_{min}}^{-1/d}u_{k,n}$ from $\partial\mathcal{X}$ (with a defined in (3.8)), it holds that any $y \in \mathcal{D} \cap B(x_i, \overline{R}_{k,n}(x_i,\delta))$ has distance larger than a from $\partial\mathcal{X}$ and hence

$$\underline{R}_{k,n}(y,\delta) \le r_{k,n}(y) \le \overline{R}_{k,n}(y,\delta), \quad y \in \mathcal{D} \cap B(x_i, \overline{R}_{k,n}(x_i,\delta)).$$

In order to prove (3.9) and (3.10) it suffices to show that $|\underline{R}_{k,n}(x_i,\delta) - \underline{R}_{k,n}(y,\delta)| \le a_{k,n}(\delta){u_{k,n}}^{1+1/d}$ and $|\overline{R}_{k,n}(x_i,\delta) - \overline{R}_{k,n}(y,\delta)| \le a_{k,n}(\delta){u_{k,n}}^{1+1/d}$, respectively. We can write $\underline{R}_{k,n}(x_i,\delta) = m \cdot f(x_i)^{-1/d}$ and $\underline{R}_{k,n}(y,\delta) = m \cdot f(y)^{-1/d}$ with $m = (1+\delta)^{-1}(1 + c_2(k/(n-1))^{1/d})^{-1}u_{k,n}$ and $\overline{R}_{k,n}(x_i,\delta) = m' \cdot f(x_i)^{-1/d}$ and $\overline{R}_{k,n}(y,\delta) = m' \cdot f(y)^{-1/d}$ with $m' = (1-\delta)^{-1}(1 - c_2(k/(n-1))^{1/d})^{-1}u_{k,n}$.

We have

$$f(y) = f(x_i) + f(x_i)\frac{f(y) - f(x_i)}{f(x_i)}.$$

Due to (3.4) and (3.5) we have

$$\left|\frac{f(y) - f(x_i)}{f(x_i)}\right| \le \frac{L\|y - x_i\|}{f_{min}} \le \frac{L\overline{R}_{k,n}(x_i,\delta)}{f_{min}} \le \frac{2Lu_{k,n}}{{f_{min}}^{1+1/d}} =: B$$

and hence $f(y) = f(x_i)(1+b)$ with $|b| \leq B < 1$ for n sufficiently large. It follows that

$$\left| \frac{1}{f(x_i)^{1/d}} - \frac{1}{f(y)^{1/d}} \right| = \left| \frac{(1+b)^{1/d} - 1}{f(x_i)^{1/d}(1+b)^{1/d}} \right| \leq \frac{1}{{f_{min}}^{1/d}} \frac{B^{1/d}}{(1-B)^{1/d}}$$

and

$$\begin{aligned} |\underline{R}_{k,n}(x_i,\delta) - \underline{R}_{k,n}(y,\delta)| &\leq \frac{m}{{f_{min}}^{1/d}} \frac{B^{1/d}}{(1-B)^{1/d}} \leq \frac{m'}{{f_{min}}^{1/d}} \frac{B^{1/d}}{(1-B)^{1/d}}, \\ |\overline{R}_{k,n}(x_i,\delta) - \overline{R}_{k,n}(y,\delta)| &\leq \frac{m'}{{f_{min}}^{1/d}} \frac{B^{1/d}}{(1-B)^{1/d}}. \end{aligned}$$

We have

$$\begin{aligned} \frac{m'}{{f_{min}}^{1/d}} \frac{B^{1/d}}{(1-B)^{1/d}} &\leq \frac{1}{{f_{min}}^{1/d}} \frac{1}{(1-\delta)(1-c_2(k/(n-1))^{1/d})} \left(\frac{2L}{{f_{min}}^{1+1/d}} \right)^{1/d} \\ &\qquad \left(1 - \frac{2Lu_{k,n}}{{f_{min}}^{1+1/d}} \right)^{-1/d} u_{k,n}^{1+1/d} \\ &=: a_{k,n}(\delta) u_{k,n}^{1+1/d}, \end{aligned}$$

where $a_{k,n}(\delta) > 0$ converges to a positive constant as $n \to \infty$ and $\delta \to 0$. ■

Due to the growth conditions on k and (3.4) we have $a_{k,n}(\delta)u_{k,n}^{1+1/d} \in o(\underline{R}_{k,n}(x_i,\delta))$ and $a_{k,n}(\delta)u_{k,n}^{1+1/d} \in o(\overline{R}_{k,n}(x_i,\delta))$ as $n \to \infty$ uniformly with respect to x_i.

The next proposition is classical. It states that with high probability in every small ball in $\mathcal{X}$ there is at least one sample point, provided n is large enough.

Proposition 3.5 (Dense sampling lemma)**.** *There exist $c_3, c_4 > 0$ such that for any $0 < \gamma \leq \varepsilon_0$ we have*

$$\forall x \in \mathcal{X} \quad \exists x_i \in \mathcal{D} : \ \|x_i - x\| \leq \gamma \tag{3.11}$$

with probability at least $1 - c_3\gamma^{-d}\exp(-c_4\gamma^d n)$.

Proof Our proof essentially coincides with the one of the sampling lemma in the supplementary material of Tenenbaum et al. (2000). It is based on a simple covering argument: We begin by covering $\mathcal{X}$ with a finite family of balls of radius $\gamma/2$. We choose the sequence of centers $p_1, p_2, \ldots$ in $\mathcal{X}$ in such a way that

$$p_{j+1} \notin \bigcup_{l=1}^{j} B_{\frac{\gamma}{2}}(p_l).$$

When this is no longer possible, we are done. The smaller balls $B_{\gamma/4}(p_j)$ are all disjoint since no two p_k and p_l, $k \neq l$, are within distance $\gamma/2$ of each other. Hence we have

$$\text{number of chosen centers} \cdot \alpha \cdot \lambda_d\left(B_{\frac{\gamma}{4}}(p_j)\right) \leq \lambda_d(\mathcal{X}),$$

where we made use of (3.3), and the number of chosen centers is bounded by

$$\text{number of chosen centers} \leq \frac{\lambda_d(\mathcal{X})}{\eta_d \left(\frac{\gamma}{4}\right)^d \alpha}. \tag{3.12}$$

In particular, this shows that our procedure of choosing centers p_j stops after a finite number of steps. Afterwards, every $x \in \mathcal{X}$ belongs to some ball $B_j := B_{\gamma/2}(p_j)$. We will show that with probability at least $1 - \frac{\lambda_d(\mathcal{X})}{\eta_d \alpha (\gamma/4)^d} \exp(-f_{min} \alpha \eta_d (\gamma/2)^d n)$ every ball B_j contains at least one sample point x_i. Since the diameter of B_j is γ, this implies (3.11).

We have

$$Pr\,(\text{no ball } B_j \text{ is empty}) = 1 - Pr\,(\text{some ball } B_j \text{ is empty}) \geq 1 - \sum_j Pr\,(B_j \text{ is empty})$$

and

$$\begin{aligned} Pr\,(B_j \text{ is empty}) = \left(1 - \int_{B_j} f(x)dx\right)^n &\leq \left(1 - f_{min}\alpha\eta_d \left(\frac{\gamma}{2}\right)^d\right)^n \\ &\leq \exp\left(-f_{min}\alpha\eta_d \left(\frac{\gamma}{2}\right)^d n\right). \end{aligned}$$

Because of (3.12) it follows that

$$Pr\,(\text{no ball } B_j \text{ is empty}) \geq 1 - \frac{\lambda_d(\mathcal{X})}{\eta_d (\frac{\gamma}{4})^d \alpha} \cdot \exp\left(-f_{min}\alpha\eta_d \left(\frac{\gamma}{2}\right)^d n\right).$$

■

We will need the two following simple inequalities.

Lemma 3.6 (Elementary inequalities). *Let $a, b \in \mathbb{R}$ with $a \geq b \geq 0$ and $d \in \mathbb{N}$. There exists a constant $C(d)$ depending only on d such that*

$$(a+b)^d \leq a^d + C(d) a^{d-1} b, \tag{3.13}$$

$$(a-b)^d \geq a^d - C(d) a^{d-1} b. \tag{3.14}$$

Proof Set $C(d) = \sum_{j=1}^d \binom{d}{j} = 2^d - 1$. We have

$$(a+b)^d = \sum_{j=0}^d \binom{d}{j} a^{d-j} b^j = a^d + \sum_{j=1}^d \binom{d}{j} a^{d-j} b^j \leq a^d + C(d) a^{d-1} b$$

and

$$(a-b)^d = \sum_{j=0}^d \binom{d}{j} a^{d-j} (-b)^j \geq a^d - \sum_{j=1}^d \binom{d}{j} a^{d-j} b^j \geq a^d - C(d) a^{d-1} b.$$

■

The next lemmas are specific to $L_{\text{DP}}(i)$ and $L_{\text{CAP}}(i)$, respectively. The following one shows that $B_{\text{SP}}(i, 2)$ indeed corresponds to the Euclidean ball $B(x_i, 2r_{k,n}(x_i))$ as we have claimed in Section 3.2.1. Recall that $i \to j$ in G if and only if $\|x_i - x_j\| \leq r_{k,n}(x_i)$.

Lemma 3.7 (Geometric argument for $L_{\mathrm{DP}}(i)$: $B_{\mathrm{SP}}(i,2)$ approximates $B(x_i, 2r_{k,n}(x_i))$**).** *Assume the events considered in Proposition 3.3 and Proposition 3.5 (with γ replaced by $\varepsilon_{k,n}$) hold. Then we have for a sample point $x_i \in \mathcal{D}$ sufficiently distant to $\partial\mathcal{X}$ and all $x_j \in \mathcal{D}$ the following implications:*

$$\|x_i - x_j\| \leq 2\underline{R}_{k,n}(x_i,\delta) - a_{k,n}(\delta)u_{k,n}^{1+1/d} - 2\varepsilon_{k,n} \quad \Rightarrow \quad d_{SP}(i,j) \leq 2, \tag{3.15}$$

$$\|x_i - x_j\| > 2\overline{R}_{k,n}(x_i,\delta) + a_{k,n}(\delta)u_{k,n}^{1+1/d} \quad \Rightarrow \quad d_{SP}(i,j) > 2. \tag{3.16}$$

Proof Assume that $\|x_i - x_j\| \leq 2\underline{R}_{k,n}(x_i,\delta) - a_{k,n}(\delta)u_{k,n}^{1+1/d} - 2\varepsilon_{k,n}$. In particular, we have $\underline{R}_{k,n}(x_i,\delta) > \varepsilon_{k,n}$. If $\|x_i - x_j\| \leq \underline{R}_{k,n}(x_i,\delta)$, either j equals i or x_j is connected to x_i according to (3.7) and hence $d_{\mathrm{SP}}(i,j) \leq 1$. If $\|x_i - x_j\| > \underline{R}_{k,n}(x_i,\delta)$, consider the point z on the line segment from x_i to x_j such that $\|x_i - z\| = \underline{R}_{k,n}(x_i,\delta) - \varepsilon_{k,n}$. Due to the assumption that (3.11) holds (with γ replaced by $\varepsilon_{k,n}$) there exists a sample point $x_l \in \mathcal{D}$ with $\|z - x_l\| \leq \varepsilon_{k,n}$. Then

$$\|x_i - x_l\| \leq \|x_i - z\| + \|z - x_l\| \leq \underline{R}_{k,n}(x_i,\delta)$$

and hence x_l is connected to x_i according to (3.7). Similarly,

$$\|x_j - x_l\| \leq \|x_j - z\| + \|z - x_l\| \leq \underline{R}_{k,n}(x_i,\delta) - a_{k,n}(\delta)u_{k,n}^{1+1/d}$$

such that x_j is connected to x_l because of $x_l \in B(x_i, \overline{R}_{k,n}(x_i,\delta))$ and (3.9). It follows that $d_{\mathrm{SP}}(i,j) \leq 2$.

If $\|x_i - x_j\| > 2\overline{R}_{k,n}(x_i,\delta) + a_{k,n}(\delta)u_{k,n}^{1+1/d}$, then (3.7) and (3.10) immediately imply that $d_{\mathrm{SP}}(i,j) > 2$. ■

Now we show that $1/L_{\mathrm{DP}}(i)$ is concentrated around 2^d, given that (3.15) and (3.16) hold.

Lemma 3.8 ($1/L_{\mathrm{DP}}(i)$ is concentrated)**.** *There exist functions $e(\delta,n)$, $E(\delta,n)$, $g(\delta,n)$, $G(\delta,n)$ with $e(\delta,n), E(\delta,n) \to 1$, $g(\delta,n), G(\delta,n) \to 0$ as $n \to \infty, \delta \to 0$ and $c_5 > 0$ such that both probabilities*

$$Pr\left(L_{\mathrm{DP}}(i)^{-1} \leq e(\delta,n)2^d - g(\delta,n) \mid x_i \in \mathcal{D} \text{ sufficiently distant to } \partial\mathcal{X}, \text{ (3.15) holds}\right), \tag{3.17}$$

$$Pr\left(L_{\mathrm{DP}}(i)^{-1} \geq E(\delta,n)2^d + G(\delta,n) \mid x_i \in \mathcal{D} \text{ sufficiently distant to } \partial\mathcal{X}, \text{ (3.16) holds}\right) \tag{3.18}$$

are upper bounded by $2\exp(-c_5\delta^2 k)$ for every $i \in \{1,\ldots,n\}$, assuming that $\varepsilon_{k,n} \in o(\underline{R}_{k,n}(x_i,\delta))$ uniformly with respect to x_i and (3.15) or (3.16) holds with probability at least 1/2 for every $x_i \in \mathcal{D}$ sufficiently distant to $\partial\mathcal{X}$.

Proof We set

$$\tilde{a} = \frac{2}{(1-\delta)(1-c_2(k/(n-1))^{1/d})} \cdot \frac{u_{k,n}}{{f_{min}}^{1/d}} + a_{k,n}(\delta)u_{k,n}^{1+1/d} \tag{3.19}$$

and

$$
\begin{aligned}
e(\delta,n) &= (1-\delta)\frac{n-1}{k+1}\left(1-\frac{L}{f_{min}}\tilde{a}\right)\eta_d\frac{u_{k,n}^{d}}{(1+\delta)^d(1+c_2(k/(n-1))^{1/d})^d},\\
E(\delta,n) &= (1+\delta)\frac{n-1}{k+1}\left(1+\frac{L}{f_{min}}\tilde{a}\right)\eta_d\frac{u_{k,n}^{d}}{(1-\delta)^d(1-c_2(k/(n-1))^{1/d})^d},\\
g(\delta,n) &= \\
&\quad (1-\delta)\frac{n-1}{k+1}\left(1-\frac{L}{f_{min}}\tilde{a}\right)\frac{\eta_d C(d)2^{d-1}f_{max}}{{f_{min}}^{1-1/d}}\frac{a_{k,n}(\delta)u_{k,n}^{d+1/d}+2\varepsilon_{k,n}u_{k,n}^{d-1}}{(1+\delta)^{d-1}(1+c_2(k/(n-1))^{1/d})^{d-1}},\\
G(\delta,n) &= \frac{1}{k+1}+\\
&\quad (1+\delta)\frac{n-1}{k+1}\left(1+\frac{L}{f_{min}}\tilde{a}\right)\frac{\eta_d C(d)2^{d-1}f_{max}}{{f_{min}}^{1-1/d}}\frac{a_{k,n}(\delta)u_{k,n}^{d+1/d}}{(1-\delta)^{d-1}(1-c_2(k/(n-1))^{1/d})^{d-1}},
\end{aligned}
$$

where $C(d)=\sum_{j=1}^{d}\binom{d}{j}$ is the constant from Lemma 3.6. It is straightforward to see that $e(\delta,n),E(\delta,n)\to 1$, $g(\delta,n),G(\delta,n)\to 0$ as $n\to\infty$, $\delta\to 0$ under the conditions on k and the assumption on $\varepsilon_{k,n}$. The proof of this lemma is very similar to the one of Proposition 3.3. We only show the bound for (3.18), the bound for (3.17) can be shown analogously.

We fix a sample point x_i with distance larger than $5{f_{min}}^{-1/d}u_{k,n}$ from the boundary of $\mathcal{X}$. In particular, x_i satisfies the requirements for being "sufficiently distant" to $\partial\mathcal{X}$ in the propositions and lemmas above. For n sufficiently large and δ sufficiently small we have $\tilde{a}\leq 5{f_{min}}^{-1/d}u_{k,n}$ and the closed ball $B=B(x_i,2\overline{R}_{k,n}(x_i,\delta)+a_{k,n}(\delta)u_{k,n}^{1+1/d})$ is fully contained in $\mathcal{X}$. Let Z_B be a random variable given as the number of points in B if $n-1$ points are drawn i.i.d. from f. Similarly as in the proof of Proposition 3.3 we obtain

$$
f(y)\leq f(x_i)\left(1+\frac{L}{f_{min}}\tilde{a}\right),\qquad f(y)\geq f(x_i)\left(1-\frac{L}{f_{min}}\tilde{a}\right),
$$

for all $y\in B$, and

$$
\begin{aligned}
\mathrm{E}(Z_B) &\leq (n-1)f(x_i)\left(1+\frac{L}{f_{min}}\tilde{a}\right)\eta_d\left(2\overline{R}_{k,n}(x_i,\delta)+a_{k,n}(\delta)u_{k,n}^{1+1/d}\right)^d,\\
&\overset{(3.13)}{\leq} (n-1)f(x_i)\left(1+\frac{L}{f_{min}}\tilde{a}\right)\eta_d\\
&\qquad \left(2^d\overline{R}_{k,n}(x_i,\delta)^d+C(d)2^{d-1}\overline{R}_{k,n}(x_i,\delta)^{d-1}a_{k,n}(\delta)u_{k,n}^{1+1/d}\right),\\
\mathrm{E}(Z_B) &\geq (n-1)f(x_i)\left(1-\frac{L}{f_{min}}\tilde{a}\right)\eta_d 2^d\overline{R}_{k,n}(x_i,\delta)^d.
\end{aligned}
$$

We have $(k+1)E(\delta,n)2^d+(k+1)G(\delta,n)-1\geq(1+\delta)\,\mathrm{E}(Z_B)$ and $\mathrm{E}(Z_B)\geq k/2$ for n sufficiently large and δ sufficiently small. It follows from the Angluin-Valiant-Inequality (see the proof of Proposition 3.3) that

$$
\begin{aligned}
&Pr(L_{\mathrm{DP}}(i)^{-1} \geq E(\delta,n)2^d + G(\delta,n) \mid (3.16) \text{ holds}) = \\
&\qquad = Pr\left(\frac{|B_{\mathrm{SP}}(i,2)|}{k+1} \geq E(\delta,n)2^d + G(\delta,n) \mid \{x_j \in \mathcal{D} : j \in B_{\mathrm{SP}}(i,2)\} \subseteq B\right) \\
&\qquad \leq 2Pr\left(\frac{|B_{\mathrm{SP}}(i,2)|}{k+1} \geq E(\delta,n)2^d + G(\delta,n) \text{ and } \{x_j \in \mathcal{D} : j \in B_{\mathrm{SP}}(i,2)\} \subseteq B\right) \\
&\qquad \leq 2Pr(Z_B \geq (k+1)E(\delta,n)2^d + (k+1)G(\delta,n) - 1) \\
&\qquad \leq 2Pr(Z_B \geq (1+\delta)\,\mathrm{E}(Z_B)) \\
&\qquad \leq 2\exp\left(-\frac{\delta^2}{3}\,\mathrm{E}(Z_B)\right) \\
&\qquad \leq 2\exp\left(-\frac{\delta^2 k}{6}\right),
\end{aligned}
$$

assuming that (3.16) holds with probability at least 1/2. ■

We want to proceed similarly for $L_{\mathrm{CAP}}(i)$. We need a generic statement about the intersection of two balls in $\mathbb{R}^d$.

Proposition 3.9 (Intersection of two balls)**.** *Consider two balls $B(M,r)$ and $B(N,s)$ in $\mathbb{R}^d$ with centers $M \neq N \in \mathbb{R}^d$ and radii $r \geq s > 0$, respectively. For their intersection $B(M,r) \cap B(N,s)$ it holds that*

$$
B(M,r) \cap B(N,s) = \begin{cases} B(N,s) \qquad \text{if } \|M-N\| \leq r-s, \\ C(M,r,r-\lambda,N) \cup C(N,s,s-\|M-N\|+\lambda,M) \\ \qquad\qquad\qquad\qquad \text{if } r-s < \|M-N\| < r+s, \\ \left\{M + r\frac{N-M}{\|M-N\|}\right\} \qquad \text{if } \|M-N\| = r+s, \\ \emptyset \qquad \text{if } \|M-N\| > r+s, \end{cases}
$$

where $\lambda = \frac{1}{2}\|M-N\| + \frac{1}{2}\frac{r^2-s^2}{\|M-N\|} > 0$ and $C(z,r,h,w)$ denotes a spherical cap of a ball $B(z,r)$ with height h $(0 \leq h \leq 2r)$ and apex on the half-line from z to w. In the second case, that is if $r-s < \|M-N\| < r+s$, the two spherical caps are disjoint except for their shared base.

Proof

- Assume that $\|M-N\| \leq r-s$. For $x \in B(N,s)$ it holds that

  $$\|x-M\| \leq \|x-N\| + \|N-M\| \leq s + r - s = r$$

  and hence $x \in B(M,r)$. This implies that $B(M,r) \cap B(N,s) = B(N,s)$.

- Assume that $r-s < \|M-N\| < r+s$. We first show that points in the intersection of the boundaries of the two balls, that is on both of the two spheres described by the equations $\|x-M\|^2 = r^2$ and $\|x-N\|^2 = s^2$, respectively, lie on an affine hyperplane that is orthogonal to the line connecting M and N: points $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ in this intersection satisfy

$$\begin{aligned}(x_1 - M_1)^2 + \ldots + (x_d - M_d)^2 = r^2,\\ (x_1 - N_1)^2 + \ldots + (x_d - N_d)^2 = s^2,\end{aligned}$$

and by subtracting the second equation from the first one we obtain

$$x_1(N_1 - M_1) + \ldots + x_d(N_d - M_d) = \frac{r^2 - s^2 - \|M\|^2 + \|N\|^2}{2}. \tag{3.20}$$

This hyperplane divides $\mathbb{R}^d$ into two halfspaces, to which we refer as upper halfspace and lower halfspace using the vector $N - M$ for orientation.

It is not difficult to see that the hyperplane described by (3.20) intersects the line connecting M and N in the point

$$P = M + \lambda \frac{N - M}{\|N - M\|},$$

where $\lambda = \frac{1}{2}\|M - N\| + \frac{1}{2}\frac{r^2 - s^2}{\|M-N\|} > 0$. It follows that the intersection of the closed upper halfspace with $B(M,r)$ is given by $C(M, r, r - \lambda, N)$ and the intersection of the closed lower halfspace with $B(N,s)$ is given by $C(N, s, s - \|M - N\| + \lambda, M)$. Clearly, these two spherical caps are disjoint except for their shared base. It remains to show that $C(M, r, r - \lambda, N)$ is fully contained in $B(N,s)$ and that $C(N, s, s - \|M - N\| + \lambda, M)$ is fully contained in $B(M,r)$: If $u \in C(M, r, r - \lambda, N)$, we can write u as $u = P + v\frac{N-M}{\|N-M\|} + h$, where $v \geq 0$ and the vector h is orthogonal to $N - M$. Since $u \in B(M,r)$, we have

$$\|u - M\|^2 = (\lambda + v)^2 + \|h\|^2 \leq r^2,$$

and it follows that

$$\begin{aligned}\|u - N\|^2 &= \left(-1 + \frac{\lambda + v}{\|N - M\|}\right)^2 \|N - M\|^2 + \|h\|^2\\ &= \|N - M\|^2 - 2(\lambda + v)\|N - M\| + (\lambda + v)^2 + \|h\|^2\\ &\leq \|N - M\|^2 - 2\lambda\|N - M\| + r^2\\ &= \|N - M\|^2 - \|N - M\|^2 - (r^2 - s^2) + r^2 = s^2.\end{aligned}$$

This shows that $C(M, r, r - \lambda, N) \subseteq B(N,s)$. Similarly, we can show that $C(N, s, s - \|M - N\| + \lambda, M) \subseteq B(M,r)$.

- Assume that $\|M - N\| = r + s$. It is straightforward to see that $M + r\frac{N-M}{\|M-N\|} \in B(M,r) \cap B(N,s)$. On the other hand, assume that $x \in B(M,r) \cap B(N,s)$. We have

$$\|M - N\| \leq \|M - x\| + \|x - N\| \leq r + s = \|M - N\| = \|M - x + x - N\|, \tag{3.21}$$

implying that $\|M - x\| = r$ and $\|x - N\| = s$. Since $\mathbb{R}^d$ equipped with the Euclidean norm is strictly convex, (3.21) also implies that there exists $c > 0$ such that $M - x = c(x - N)$. It is easy to see that $x = M + r\frac{N-M}{\|M-N\|}$.

- Assume that $\|M-N\| > r+s$. Assume that x was an element of $B(M,r) \cap B(N,s)$. Then we would have $\|x-M\| \leq r$ and $\|x-N\| \leq s$, which yields the contradiction
$$\|M-N\| \leq \|M-x\| + \|x-N\| \leq r+s.$$

■

The following Lemmas 3.10 and 3.11 are similar to Lemmas 3.7 and 3.8, respectively.

Lemma 3.10 (Geometric argument for $L_{\text{CAP}}(i)$: $B_{\text{SP}}(i,1) \cap B_{\text{SP}}(j_0,1)$ approximates union of spherical caps)**.** *Assume the event considered in Proposition 3.3 holds. Then we have for a sample point $x_i \in \mathcal{D}$ sufficiently distant to $\partial\mathcal{X}$ and all $j \in \{1,\ldots,n\}$ with $i \to j$ the following implications:*

$$\begin{aligned} \|x_i - x_j\| \leq a_{k,n}(\delta) u_{k,n}^{1+1/d} \quad &\Rightarrow \quad B(x_i,\underline{R}) \cap B(x_j,\underline{T}) = B(x_j,\underline{T}), && (3.22)\\ \|x_i - x_j\| > a_{k,n}(\delta) u_{k,n}^{1+1/d} \quad &\Rightarrow \quad B(x_i,\underline{R}) \cap B(x_j,\underline{T}) \supseteq \\ &\qquad C(x_i,\underline{T},h(x_i,x_j),x_j) \cup C(x_j,\underline{T},h(x_i,x_j),x_i), && (3.23) \end{aligned}$$

where $h(x_i,x_j) = \underline{T} - \frac{1}{2}\|x_i - x_j\|$ and we abbreviate

$$\underline{R} = \underline{R}_{k,n}(x_i,\delta), \qquad \underline{T} = \underline{R} - a_{k,n}(\delta) u_{k,n}^{1+1/d}.$$

As in Proposition 3.9, $C(z,r,h,w)$ denotes a spherical cap of a ball $B(z,r)$ with height h $(0 \leq h \leq 2r)$ and apex on the half-line from z to w. In (3.23) the two spherical caps are disjoint except for their shared base.

Furthermore, additionally assuming that the event considered in Proposition 3.5 with γ replaced by $\varepsilon_{k,n}$ holds and $\varepsilon_{k,n} \in o(\underline{R}_{k,n}(x_i,\delta))$ uniformly with respect to x_i, there exists a sample point x_l with $i \to l$ such that

$$B(x_i,\overline{R}) \cap B(x_l,\overline{T}) \subseteq C(x_i,\overline{T},h,x_l) \cup C(x_l,\overline{T},h,x_i), \tag{3.24}$$

where $h = \overline{T} - \frac{1}{2}(\underline{R} - 2\varepsilon_{k,n})$ and we abbreviate

$$\overline{R} = \overline{R}_{k,n}(x_i,\delta), \qquad \overline{T} = \overline{R} + a_{k,n}(\delta) u_{k,n}^{1+1/d}.$$

Proof If $i \to j$, we have $\|x_i - x_j\| \leq \overline{R}_{k,n}(x_i,\delta)$ according to (3.7). For n sufficiently large and δ sufficiently small it holds that

$$\overline{R}_{k,n}(x_i,\delta) < 2\underline{R}_{k,n}(x_i,\delta) - 2a_{k,n}(\delta) u_{k,n}^{1+1/d} = 2\underline{T}$$

for all x_i sufficiently distant to $\partial\mathcal{X}$, and the implications (3.22) and (3.23) immediately follow from $B(x_i,\underline{R}) \cap B(x_j,\underline{T}) \supseteq B(x_i,\underline{T}) \cap B(x_j,\underline{T})$ and Proposition 3.9.

Assuming that $\varepsilon_{k,n} \in o(\underline{R}_{k,n}(x_i,\delta))$ uniformly with respect to x_i we have $2\varepsilon_{k,n} < \underline{R}$ for n sufficiently large. Let $z \in \mathbb{R}^d$ be any point with $\|x_i - z\| = \underline{R} - \varepsilon_{k,n}$. Due to the assumption that (3.11) holds (with γ replaced by $\varepsilon_{k,n}$) there exists a sample point $x_l \in \mathcal{D}$ with $\|z - x_l\| \leq \varepsilon_{k,n}$. It follows that

$$\underline{R} - 2\varepsilon_{k,n} \leq \|x_i - x_l\| \leq \underline{R}, \tag{3.25}$$

and according to (3.7) we have $i \to l$. Because of $B(x_i, \overline{R}) \cap B(x_l, \overline{T}) \subseteq B(x_i, \overline{T}) \cap B(x_l, \overline{T})$, (3.24) immediately follows from Proposition 3.9 and (3.25). ■

Lemma 3.11 ($L_{\text{CAP}}(i)$ is concentrated)**.** *There exist functions $e'(\delta, n)$, $E'(\delta, n)$, $g'(\delta, n)$, $G'(\delta, n)$ with $e'(\delta, n), E'(\delta, n) \to 1$, $g'(\delta, n), G'(\delta, n) \to 0$ as $n \to \infty, \delta \to 0$ and $c_6 > 0$ such that the probabilities*

$$\begin{aligned} Pr\big(L_{\text{CAP}}(i) \leq e'(\delta, n)S(d) - g'(\delta, n) \mid\ & x_i \in \mathcal{D} \textit{ sufficiently distant to } \partial\mathcal{X}\textit{, the event} \\ & \textit{considered in Proposition 3.3 holds, } \forall j \in \{1, \ldots, n\} \textit{ with } i \to j \textit{ the} \\ & \textit{implications } (3.22) \textit{ and } (3.23) \textit{ hold}\big), \end{aligned} \tag{3.26}$$

$$\begin{aligned} Pr\big(L_{\text{CAP}}(i) \geq E'(\delta, n)S(d) + G'(\delta, n) \mid\ & x_i \in \mathcal{D} \textit{ sufficiently distant to } \partial\mathcal{X}\textit{, the event} \\ & \textit{considered in Proposition 3.3 holds, } \exists x_l \in \mathcal{D} \textit{ with } i \to l \textit{ and } (3.24)) \end{aligned} \tag{3.27}$$

are upper bounded by $2k\exp(-c_6\delta^2 k)$ and $2\exp(-c_6\delta^2 k)$, respectively, for every $i \in \{1, \ldots, n\}$, assuming that $\varepsilon_{k,n} \in o(\underline{R}_{k,n}(x_i, \delta))$ uniformly with respect to x_i and the probability of the event considered in Proposition 3.3 is larger than 1/2.

Proof According to Li (2011), the volume $\lambda_d(C(z, r, h, w))$ of a spherical cap of a ball $B(z, r) \subseteq \mathbb{R}^d$ with height h is given by (independently of the location of the apex)

$$\lambda_d(C(z, r, h, w)) = \begin{cases} \frac{1}{2}\eta_d r^d I_{\frac{2rh-h^2}{r^2}}\left(\frac{d+1}{2}, \frac{1}{2}\right) & \text{if } 0 \leq h \leq r, \\ \eta_d r^d - \frac{1}{2}\eta_d r^d I_{\frac{2rh-h^2}{r^2}}\left(\frac{d+1}{2}, \frac{1}{2}\right) & \text{if } r < h \leq 2r. \end{cases} \tag{3.28}$$

$I_x(a, b)$ for $0 \leq x \leq 1$ and $a, b > 0$ denotes the regularized incomplete beta function, which is given by

$$I_x(a, b) = \frac{1}{B(a, b)} \int_0^x t^{a-1}(1-t)^{b-1}\, dt,$$

where $B(a, b) > 0$ denotes the beta function. Note that $I_x(a, b)$ as a function of x is monotonically increasing with $I_0(a, b) = 0$ and $I_1(a, b) = 1$, and recall that $S(d) = I_{3/4}((d + 1)/2, 1/2)$. We have $0 < S(d) < 1$ for $d \geq 0$. For $-3/4 \leq x \leq 1/4$ we set

$$F(x) = \frac{1}{B((d+1)/2, 1/2)} \int_{\min\{\frac{3}{4}, \frac{3}{4}+x\}}^{\max\{\frac{3}{4}, \frac{3}{4}+x\}} t^{(d+1)/2-1}(1-t)^{-1/2}\, dt.$$

We have $F(x) \geq 0$, $F(0) = 0$, $F(x) \to 0$ as $x \to 0$, and $F(x_1) \leq F(x_2)$ for $0 \leq x_1 \leq x_2 \leq 1/4$ and $F(x_1) \geq F(x_2)$ for $-3/4 \leq x_1 \leq x_2 \leq 0$.

We set

$$\begin{aligned}
t(\delta,n) &= 1 - 4{f_{max}}^{1/d}a_{k,n}(\delta)u_{k,n}^{1/d} - \frac{(1+\delta)^2(1+c_2(k/(n-1))^{1/d})^2}{(1-\delta)^2(1-c_2(k/(n-1))^{1/d})^2},\\
T(\delta,n) &= {f_{max}}^{1/d}a_{k,n}(\delta)u_{k,n}^{1/d} + \frac{{f_{max}}^{1/d}\varepsilon_{k,n}}{u_{k,n}} + \\
&\quad \frac{{f_{max}}^{1/d}}{{f_{min}}^{1/d}}\left(\frac{1}{(1-\delta)(1-c_2(k/(n-1))^{1/d})} - \frac{1}{(1+\delta)(1+c_2(k/(n-1))^{1/d})}\right)
\end{aligned}$$

and

$$\begin{aligned}
e'(\delta,n) &= (1-\delta)\frac{n-2}{k+1}\left(1-\frac{L}{f_{min}}\tilde{a}\right)\eta_d\frac{u_{k,n}^d}{(1+\delta)^d(1+c_2(k/(n-1))^{1/d})^d},\\
E'(\delta,n) &= (1+\delta)\frac{n-2}{k+1}\left(1+\frac{L}{f_{min}}\tilde{a}\right)\eta_d\frac{u_{k,n}^d}{(1-\delta)^d(1-c_2(k/(n-1))^{1/d})^d},\\
g'(\delta,n) &= (1-\delta)\frac{n-2}{k+1}\left(1-\frac{L}{f_{min}}\tilde{a}\right)\eta_d\frac{u_{k,n}^d}{(1+\delta)^{d-1}(1+c_2(k/(n-1))^{1/d})^{d-1}}\\
&\qquad\qquad \left[F(t(\delta,n)) + \frac{f_{max}}{{f_{min}}^{1-1/d}}C(d)a_{k,n}(\delta)u_{k,n}^{1/d}\right],\\
G'(\delta,n) &= \frac{2}{k+1} + E'(\delta,n)\left[F\left(T(\delta,n)\right) + \frac{f_{max}}{{f_{min}}^{1-1/d}}C(d)a_{k,n}(\delta)u_{k,n}^{1/d}\right],
\end{aligned}$$

where $C(d) = \sum_{j=1}^{d}\binom{d}{j}$ as in Lemma 3.6 and $\tilde{a}$ is defined in (3.19). It is straightforward to see that $e'(\delta,n), E'(\delta,n) \to 1$, $g'(\delta,n), G'(\delta,n) \to 0$ as $n \to \infty, \delta \to 0$ under the conditions on k and the assumption on $\varepsilon_{k,n}$.

As in the proof of Lemma 3.8 we fix a sample point x_i with distance larger than $5{f_{min}}^{-1/d}u_{k,n}$ from the boundary of $\mathcal{X}$. Then for every z in $B(x_i,\overline{R})$ the closed ball $B(z,\overline{T})$ is fully contained in $B = B(x_i, 2\overline{R}_{k,n}(x_i,\delta) + a_{k,n}(\delta)u_{k,n}^{1+1/d}) \subseteq \mathcal{X}$, where we use the notation introduced in Lemma 3.10. From the proof of Lemma 3.8 we obtain

$$f(y) \le f(x_i)\left(1+\frac{L}{f_{min}}\tilde{a}\right), \qquad f(y) \ge f(x_i)\left(1-\frac{L}{f_{min}}\tilde{a}\right), \tag{3.29}$$

for all $y \in B$.

We begin with showing the upper bound for (3.27). We have

$$\begin{aligned}
&Pr\left(L_{\mathrm{CAP}}(i) \ge E'(\delta,n)S(d) + G'(\delta,n) \mid \text{the event considered in Proposition 3.3 holds,}\right.\\
&\qquad\qquad \left.\exists x_l \in \mathcal{D} \text{ with } i \to l \text{ and (3.24)}\right)\\
&\quad \le Pr\left(|B_{\mathrm{SP}}(i,1) \cap B_{\mathrm{SP}}(l,1)|/(k+1) \ge E'(\delta,n)S(d) + G'(\delta,n) \mid \text{the event}\right.\\
&\qquad\qquad\qquad \left.\text{considered in Proposition 3.3 holds, } \exists x_l \in \mathcal{D} \text{ with } i \to l \text{ and (3.24)}\right).
\end{aligned}$$

After fixing a sample point x_l with $\|x_i - x_l\| \leq \overline{R}$, $i \to l$, and (3.24) we obtain

$$\begin{aligned} &Pr\left(|B_{\mathrm{SP}}(i,1) \cap B_{\mathrm{SP}}(l,1)| \geq (k+1)E'(\delta,n)S(d) + (k+1)G'(\delta,n) \mid \text{the event} \right. \\ &\qquad\qquad \left. \text{considered in Proposition 3.3 holds}\right) \\ &\quad \leq 2Pr\left(|B_{\mathrm{SP}}(i,1) \cap B_{\mathrm{SP}}(l,1)| \geq (k+1)E'(\delta,n)S(d) + (k+1)G'(\delta,n) \text{ and} \right. \\ &\qquad\qquad\qquad \left. r_{k,n}(x_i) \leq \overline{R}, r_{k,n}(x_l) \leq \overline{T}\right), \end{aligned}$$

where we assume that the probability of the event considered in Proposition 3.3 is larger than 1/2.

Let Z_C be a random variable given as the number of points in $C = C(x_i, \overline{T}, h, x_l) \cup C(x_l, \overline{T}, h, x_i) \subseteq B$ if $n-2$ points are drawn i.i.d. from f. We have

$$\begin{aligned} &Pr\left(|B_{\mathrm{SP}}(i,1) \cap B_{\mathrm{SP}}(l,1)| \geq (k+1)E'(\delta,n)S(d) + (k+1)G'(\delta,n) \text{ and} \right. \\ &\qquad\qquad\qquad \left. r_{k,n}(x_i) \leq \overline{R}, r_{k,n}(x_l) \leq \overline{T}\right) \\ &\qquad \leq Pr(Z_C \geq (k+1)E'(\delta,n)S(d) + (k+1)G'(\delta,n) - 2). \end{aligned}$$

Note that $h = \overline{T} - \frac{1}{2}(\underline{R} - 2\varepsilon_{k,n}) \leq \overline{T}$. Due to (3.28) and (3.29) we have

$$\begin{aligned} \mathrm{E}(Z_C) &\leq (n-2)f(x_i)\left(1 + \frac{L}{f_{min}}\tilde{a}\right)\eta_d \overline{T}^d I_{\frac{\overline{T}^2 - \underline{R}^2/4 + \underline{R}\varepsilon_{k,n} - \varepsilon_{k,n}^2}{\overline{T}^2}}\left(\frac{d+1}{2}, \frac{1}{2}\right) \\ &= (n-2)f(x_i)\left(1 + \frac{L}{f_{min}}\tilde{a}\right)\eta_d \overline{T}^d \left[S(d) + F\left(\frac{(\overline{T}^2 - \underline{R}^2)/4 + \underline{R}\varepsilon_{k,n} - \varepsilon_{k,n}^2}{\overline{T}^2}\right)\right] \\ &\overset{(3.13)}{\leq} (n-2)f(x_i)\left(1 + \frac{L}{f_{min}}\tilde{a}\right)\eta_d \left[\overline{R}^d + C(d)\overline{R}^{d-1} a_{k,n}(\delta) u_{k,n}^{1+1/d}\right] \\ &\qquad \left[S(d) + F\left(\frac{(\overline{T}^2 - \underline{R}^2)/4 + \underline{R}\varepsilon_{k,n} - \varepsilon_{k,n}^2}{\overline{T}^2}\right)\right] \end{aligned}$$

and

$$\mathrm{E}(Z_C) \geq (n-2)f(x_i)\left(1 - \frac{L}{f_{min}}\tilde{a}\right)\frac{1}{2}\eta_d \overline{T}^d S(d).$$

It is straightforward to see that $(k+1)E'(\delta,n)S(d) + (k+1)G'(\delta,n) - 2 \geq (1+\delta)\,\mathrm{E}(Z_C)$ and $\mathrm{E}(Z_C) \geq S(d)k/4$ for n sufficiently large and δ sufficiently small, and from here the proof is analogous to the one of Lemma 3.8.

We now show the upper bound for (3.26). We have

$$\begin{aligned} &Pr\left(L_{\mathrm{CAP}}(i) \leq e'(\delta,n)S(d) - g'(\delta,n) \mid \text{the event considered in Proposition 3.3 holds,} \right. \\ &\qquad\qquad \left. \forall j \in \{1,\ldots,n\} \text{ with } i \to j \text{ the implications (3.22) and (3.23) hold}\right) \\ &\leq \sum_{m=1}^{k} Pr\left(|B_{\mathrm{SP}}(i,1) \cap B_{\mathrm{SP}}(i(m),1)|/(k+1) \leq e'(\delta,n)S(d) - g'(\delta,n) \mid \text{the event} \right. \\ &\qquad\qquad\qquad \text{considered in Proposition 3.3 holds, } \forall j \in \{1,\ldots,n\} \text{ with } i \to j \\ &\qquad\qquad\qquad\qquad \left. \text{the implications (3.22) and (3.23) hold}\right), \end{aligned}$$

where $x_{i(m)}$ denotes the m-th nearest neighbor of x_i. We have

$$\begin{aligned}
&Pr\left(|B_{\mathrm{SP}}(i,1)\cap B_{\mathrm{SP}}(i(m),1)|/(k+1)\le e'(\delta,n)S(d)-g'(\delta,n)\mid \text{the event considered}\right.\\
&\quad\text{in Proposition 3.3 holds, } \forall j \text{ with } i\to j \text{ the implications (3.22) and (3.23) hold}\Big)\\
&\quad\le 2Pr\left(|B_{\mathrm{SP}}(i,1)\cap B_{\mathrm{SP}}(i(m),1)|/(k+1)\le e'(\delta,n)S(d)-g'(\delta,n) \text{ and}\right.\\
&\qquad r_{k,n}(x_i)\ge \underline{R}, r_{k,n}(x_{i(m)})\ge \underline{T}\mid \|x_i-x_{i(m)}\|\le \overline{R}, \text{ the implications (3.22)}\\
&\qquad\qquad \left.\text{and (3.23) hold for } x_i \text{ and } x_j=x_{i(m)}\right),
\end{aligned}$$

where we assume that the probability of the event considered in Proposition 3.3 is larger than 1/2.

After fixing a sample point $x_{i(m)}$ with $\|x_i-x_{i(m)}\|\le \overline{R}$ and such that the implications (3.22) and (3.23) hold for x_i and $x_j=x_{i(m)}$ we obtain

$$\begin{aligned}
&Pr\left(|B_{\mathrm{SP}}(i,1)\cap B_{\mathrm{SP}}(i(m),1)|/(k+1)\le e'(\delta,n)S(d)-g'(\delta,n) \text{ and}\right.\\
&\qquad\qquad \left. r_{k,n}(x_i)\ge \underline{R}, r_{k,n}(x_{i(m)})\ge \underline{T}\right)\\
&\quad\le Pr(Z_C\le (k+1)e'(\delta,n)S(d)-(k+1)g'(\delta,n)),
\end{aligned}$$

where Z_C is a random variable given as the number of points in

$$C=\begin{cases} B(x_{i(m)},\underline{T}) & \text{if } \|x_i-x_{i(m)}\|\le a_{k,n}(\delta)u_{k,n}^{1+1/d},\\ C(x_i,\underline{T},h(x_i,x_{i(m)}),x_{i(m)})\cup C(x_{i(m)},\underline{T},h(x_i,x_{i(m)}),x_i) & \text{otherwise}\end{cases}$$

if $n-2$ points are drawn i.i.d. from f. Recall from Lemma 3.10 that in the second case the two spherical caps are disjoint except for their shared base and note that $C\subseteq B$. It is elementary to show that in both cases $(1-\delta)\,\mathrm{E}(Z_C)\ge (k+1)e'(\delta,n)S(d)-(k+1)g'(\delta,n)$ and $\mathrm{E}(Z_C)\ge S(d)k/4$ for n sufficiently large and δ sufficiently small (in the second case we have to distinguish the two cases $a_{k,n}(\delta)u_{k,n}^{1+1/d}<\|x_i-x_{i(m)}\|\le \underline{T}$ and $\underline{T}<\|x_i-x_{i(m)}\|\le \overline{R}$). Similarly as in the proof of Proposition 3.3 or Lemma 3.8 we obtain

$$Pr(Z_C\le (k+1)e'(\delta,n)S(d)-(k+1)g'(\delta,n))\le \exp\left(-\frac{S(d)\delta^2 k}{8}\right),$$

which implies that (3.26) is upper bounded by $2k\exp(-S(d)\delta^2k/8)$. ■

Now we can prove Theorem 3.2.

Proof of Theorem 3.2: We begin with showing convergence of $L_{\mathrm{DP}}(i)$ and $L_{\mathrm{CAP}}(i)$ for every prespecified $i\in\{1,\dots,n\}$. We choose $\varepsilon_{k,n}=(k'/n)^{1/d}$ for some $k'=k'(n)$ satisfying $k'\in o(k)$ and $\log n\in o(k')$. In particular, this implies $\varepsilon_{k,n}\in o(\underline{R}_{k,n}(x_i,\delta))$ as $n\to\infty$ uniformly with respect to x_i due to (3.4). In all the statements above, the assumption of the sample point $x_i\in\mathcal{D}$ being "sufficiently distant" to $\partial\mathcal{X}$ is satisfied whenever x_i has distance larger than $5{f_{min}}^{-1/d}u_{k,n}$ from $\partial\mathcal{X}$. This quantity tends to 0 as $n\to\infty$. Due to (3.2) and a probability measure being continuous from above, it follows that there exists $b(n)$ with $b(n)\to 0$ as $n\to\infty$ such that the probability of x_i being not sufficiently distant to $\partial\mathcal{X}$ is upper bounded by $b(n)$. Applying a simple union bound, we see that with probability at least $1 - 2n\exp(-c_1\delta^2 k) - c_3(n/k')\exp(-c_4 k') - b(n)$ both the events of Proposition 3.3 and Proposition 3.5 (with γ replaced by $\varepsilon_{k,n}$) hold and x_i is sufficiently distant to $\partial\mathcal{X}$. It follows from Lemma 3.8 that

$$\begin{aligned} &Pr\left(e(\delta,n)2^d - g(\delta,n) \le L_{\mathrm{DP}}(i)^{-1} \le E(\delta,n)2^d + G(\delta,n)\right) \ge \\ &\qquad 1 - 4\exp(-c_5\delta^2 k) - 4n\exp(-c_1\delta^2 k) - 2c_3\frac{n}{k'}\exp(-c_4 k') - 2b(n). \end{aligned}$$

Similarly, it follows from Lemma 3.11 that

$$\begin{aligned} &Pr\left(e'(\delta,n)S(d) - g'(\delta,n) \le L_{\mathrm{CAP}}(i) \le E'(\delta,n)S(d) + G'(\delta,n)\right) \ge \\ &\qquad 1 - 4k\exp(-c_6\delta^2 k) - 4n\exp(-c_1\delta^2 k) - 2c_3\frac{n}{k'}\exp(-c_4 k') - 2b(n). \end{aligned}$$

Both expressions on the right-hand side tend to 1 as $n \to \infty$ for fixed δ, which shows that $1/L_{\mathrm{DP}}(i)$ converges in probability to 2^d and $L_{\mathrm{CAP}}(i)$ converges in probability to $S(d)$. Since $x \mapsto 1/x$ is continuous on $\mathbb{R}_{>0}$, it follows that $L_{\mathrm{DP}}(i)$ converges to $1/2^d$.

It remains to show that the averaged statistics $L_{\mathrm{DP}}(A)$ and $L_{\mathrm{CAP}}(A)$, for any prespecified $A \subseteq \{1,\ldots,n\}$, converge in probability to $1/2^d$ and $S(d)$, respectively. We only prove the claim for $L_{\mathrm{DP}}(A)$. Proving the claim for $L_{\mathrm{CAP}}(A)$ works in the same way.

We have $0 \le L_{\mathrm{DP}}(i) = |B_{\mathrm{SP}}(i,1)|/|B_{\mathrm{SP}}(i,2)| \le 1$, and this implies that $L_{\mathrm{DP}}(i)$ does not only converge in probability to $1/2^d$, but also in quadratic mean (e.g., Hajek, 2015, Proposition 2.7). In particular, the expectation of $L_{\mathrm{DP}}(i)$ converges to $1/2^d$ and its variance tends to zero, that is

$$\mathrm{E}(L_{\mathrm{DP}}(i)) \to \frac{1}{2^d}, \quad \mathrm{Var}(L_{\mathrm{DP}}(i)) \to 0 \ \text{ as } \ n \to \infty.$$

The distribution of $L_{\mathrm{DP}}(i)$ is the same for every $i \in \{1,\ldots,n\}$, and it follows that

$$\begin{aligned} \mathrm{E}(L_{\mathrm{DP}}(A)) &= \frac{1}{|A|}\sum_{i\in A}\mathrm{E}(L_{\mathrm{DP}}(i)) = \frac{1}{|A|}\sum_{i\in A}\mathrm{E}(L_{\mathrm{DP}}(1)) = \mathrm{E}(L_{\mathrm{DP}}(1)) \to \frac{1}{2^d}, \\ \mathrm{Var}(L_{\mathrm{DP}}(A)) &\le \frac{1}{|A|^2}\left(\sum_{i\in A}\mathrm{Var}(L_{\mathrm{DP}}(i)) + \sum_{i\neq j\in A}\sqrt{\mathrm{Var}(L_{\mathrm{DP}}(i))}\sqrt{\mathrm{Var}(L_{\mathrm{DP}}(j))}\right) \\ &= \frac{1}{|A|^2}\left(\sum_{i\in A}\mathrm{Var}(L_{\mathrm{DP}}(1)) + \sum_{i\neq j\in A}\sqrt{\mathrm{Var}(L_{\mathrm{DP}}(1))}\sqrt{\mathrm{Var}(L_{\mathrm{DP}}(1))}\right) \\ &= \mathrm{Var}(L_{\mathrm{DP}}(1)) \to 0. \end{aligned}$$

This implies that $L_{\mathrm{DP}}(A)$ converges in quadratic mean to $1/2^d$ and hence also in probability (e.g., Hajek, 2015, Proposition 2.7). ■

3.4 Experiments

In this section we present a number of experiments to evaluate our estimators. In particular, these experiments confirm our finding of Section 3.2.3 that E_{CAP} performs better than E_{DP} and should be preferred in practice.

3.4.1 A first comparison with estimators from the literature

To get a first impression, we compared our estimators $E_{\text{DP}}(V)$ and $E_{\text{CAP}}(V)$ to three standard estimators from the literature, all of them relying on cardinal distance information: The recent estimator MLE of Levina and Bickel (2004), which seems to be state of the art, and two widely used classical estimators: the correlation dimension-estimator CorrDim (Grassberger and Procaccia, 1983) and the estimator RegDim. MLE is the average of estimators $\hat{m}_k$ for several values of $k \in \mathbb{N}$, where $\hat{m}_k$ in turn is the average of local maximum likelihood estimators $\hat{m}_k(x_i)$ that estimate the intrinsic dimension of the data set around a data point x_i based on the distances between x_i and its k nearest neighbors. CorrDim estimates the dimension by regressing $\log C_r$ on $\log r$ over a suitable range of $r > 0$, where

$$C_r = \frac{2}{n(n-1)} \sum_{i=1}^{n} \sum_{j=i+1}^{n} \mathbb{1}\left\{\|x_i - x_j\|_{\mathbb{R}^D} \leq r\right\}$$

is the normalized number of pairs of data points with distance not more than r from each other. Similarly, RegDim works by regressing $\log R_k$ on $\log k$, where

$$R_k = \frac{1}{n} \sum_{i=1}^{n} r_{k,n}(x_i)$$

is the average k-NN radius of the data points. This is a slightly simplified version of the algorithm suggested by Pettis et al. (1979). Since MLE, CorrDim, and RegDim (and also E_{DP}) yield real-valued estimates of the intrinsic dimension, we implemented E_{CAP} as to provide a real-valued estimate too. To this end we used a lookup table of argument-value pairs of S with mesh width 0.01 (compare with Section 3.2.4).

For several artificial and real data sets, Table 3.1 shows the estimated dimensions for the various estimators. All considered estimators require to set some parameters: a single parameter k for E_{DP} and E_{CAP} and two parameters k_1, k_2 for MLE, CorrDim, and RegDim. For MLE these parameters determine the range for averaging over $\hat{m}_k$ and for CorrDim and RegDim the range for regressing (for CorrDim the range is given as $[r_{k_1}, \ldots, r_{k_2}]$, where r_i denotes the i-th smallest entry in the distance matrix of the data set). For all experiments except (9) we set the parameters for MLE and RegDim as $k_1 = 10$, $k_2 = 20$ and for CorrDim as $k_1 = 10$, $k_2 = 100$ like Levina and Bickel (2004), who also performed the experiments (2), (8), and (9). In experiment (9), like Levina and Bickel, we changed the parameters for CorrDim to $k_1 = 500$, $k_2 = 1000$ since the original choice leads to the obviously wrong result of an estimated dimension of 19.7. For E_{DP} and E_{CAP} we simply set $k = 15$ if the size of the data set is less than or equal to 1000 and $k = 20$ otherwise (in which case the size of the data set is 5000 or slightly greater).

For the artificial data sets the interpretation of the results is straightforward. The naive estimator $E_{\text{DP}}(V)$ only gives reasonable results for the experiments (1) and (2), where the dimension of the data set is small. It is highly biased in the higher-dimensional cases. This confirms our arguments of Section 3.2.3. The estimator $E_{\text{CAP}}(V)$ performs comparably to the estimators MLE, CorrDim, and RegDim. This is quite surprising, given that $E_{\text{CAP}}(V)$ only gets the directed, but unweighted k-NN graph on a data set as input, whereas MLE, CorrDim, and RegDim get to see cardinal distance information.

Table 3.1: Estimated dimensions for several data sets. n denotes the size of the data set and d its true dimension.

				Our estimators (k-NN graph)		**Standard estimators (distance values)**		
	n	**Distribution / Data set**	d	$E_{\mathrm{CAP}}(V)$	$E_{\mathrm{DP}}(V)$	MLE	CorrDim	RegDim
	Artificial data sets (results averaged over 100 runs, $\pm STD$)							
(1)	1000	uniform on a helix in $\mathbb{R}^3$	1	1.00 ($\pm$0.05)	0.88 ($\pm$0.01)	1.00 ($\pm$0.01)	1.00 ($\pm$0.11)	0.99 ($\pm$0.01)
(2)	1000	Swiss roll in $\mathbb{R}^3$	2	2.14 ($\pm$0.05)	1.44 ($\pm$0.01)	1.94 ($\pm$0.02)	1.99 ($\pm$0.23)	1.87 ($\pm$0.04)
(3)	1000	$N_5(0, I)$	5	5.33 ($\pm$0.07)	2.47 ($\pm$0.01)	5.00 ($\pm$0.04)	4.91 ($\pm$0.56)	4.86 ($\pm$0.05)
(4)	1000	uniform on sphere $\mathbb{S}^7 \subseteq \mathbb{R}^8$	7	5.88 ($\pm$0.06)	2.82 ($\pm$0.01)	6.53 ($\pm$0.07)	6.85 ($\pm$0.66)	6.23 ($\pm$0.09)
(5)	5000	uniform on sphere $\mathbb{S}^7 \subseteq \mathbb{R}^8$	7	6.85 ($\pm$0.03)	3.21 ($\pm$0.00)	6.72 ($\pm$0.03)	6.95 ($\pm$0.98)	6.46 ($\pm$0.04)
(6)	1000	uniform on $[0, 1]^{12}$	12	7.74 ($\pm$0.08)	3.04 ($\pm$0.01)	9.32 ($\pm$0.10)	10.66 ($\pm$1.18)	8.78 ($\pm$0.10)
(7)	5000	uniform on $[0, 1]^{12}$	12	9.24 ($\pm$0.04)	3.50 ($\pm$0.00)	9.76 ($\pm$0.05)	10.83 ($\pm$1.49)	9.26 ($\pm$0.05)
	Real data sets (D = dimension of observation space)							
(8)	698	Isomap faces, $D = 4096 = 64^2$	?	3.04	1.73	3.99	3.53	4.22
(9)	481	Hands, $D = 245760$	?	1.27	0.95	2.88	3.92	2.56
(10)	7141	MNIST digit 3, $D = 784 = 28^2$	?	8.92	3.21	15.95	14.17	14.75
(11)	6824	MNIST digit 4, $D = 784 = 28^2$	?	8.13	3.07	14.44	9.54	13.16
(12)	6313	MNIST digit 5, $D = 784 = 28^2$	?	8.40	3.12	15.55	18	14.28

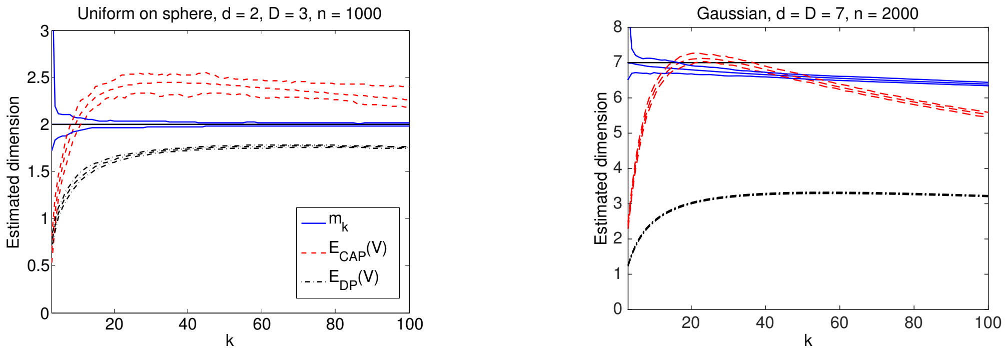


Figure 3.4: The estimates from $\hat{m}_k$, $E_{\mathrm{DP}}(V)$, and $E_{\mathrm{CAP}}(V)$ as a function of k for 1000 points from a uniform distribution on the hypersphere $\mathbb{S}^2 \subseteq \mathbb{R}^3$ (left) and for 2000 points from a 7-dim Gaussian $N_7(0, I)$ (right). The curves show the average over 100 runs of the experiment together with the minimum and the maximum of the 100 runs. The solid black lines show the true dimension.

For the real data sets the interpretation is not so obvious since the true intrinsic dimensions are unknown. Although the Isomap faces data set, consisting of images of the face of a sculpture observed under different pose and lighting conditions, is usually considered to be 3-dimensional, Levina and Bickel (2004) argue that its dimension should be higher because of the fact that we only deal with 2-dimensional projections of the face. Similarly, according to Levina and Bickel the intrinsic dimension of the Hands data set, which is a sequence of snapshots of a hand moving along a one-dimensional curve, should be higher than one. In any case, the results of $E_{\mathrm{CAP}}(V)$ do not seem to be unreasonable, in particular if one additionally compares them to the results obtained by Hein and Audibert (2005). In their experiments, Hein and Audibert provide a dimension estimate of three for the Isomap faces data set and estimates of 14, 13, and 12 for the sets of MNIST digits 3, 4, and 5, respectively.

3.4.2 Our estimators in detail

We performed experiments to investigate in artificial data sets how our estimators behave with respect to the choice of the parameter k, the sample size n, the true intrinsic dimension d, the presence of noise, and the size of A. As competitor we chose the state-of-the-art estimator $\hat{m}_k$. Note that $\hat{m}_k$ is also based on the k nearest neighbors of data points, but explicitly uses distance values. Because our estimators do not get any cardinal distance information, we cannot expect E_{DP} and E_{CAP} to perform as well as $\hat{m}_k$, but consider the latter as a benchmark. In their paper, Levina and Bickel (2004) suggest to average over $\hat{m}_k$ for a range of k (yielding the estimator MLE) in order to reduce the risk of choosing a bad value for it. In principle, this could also be done with our estimators, but in our setting this would require additional ordinal data as input, and so we do not want to pursue this idea any further.

Dependence on k

Figure 3.4 shows the estimates obtained from the estimators $\hat{m}_k$, $E_{\mathrm{DP}}(V)$, and $E_{\mathrm{CAP}}(V)$ as a function of the parameter k. In the experiment shown in the left plot the data set

Table 3.2: Estimated dimensions for n points from a 7-dim Gaussian $N_7(0, I)$ (average over 10 data sets, $\pm STD$). R denotes a random choice (without replacement) of 10 vertices.

	$E_{\text{CAP}}(R)$	$E_{\text{DP}}(R)$
$n = 5 \cdot 10^4$, $k = 500$	6.77 ($\pm$0.19)	4.36 ($\pm$0.01)
$n = 5 \cdot 10^5$, $k = 1000$	7.58 ($\pm$0.12)	5.01 ($\pm$0.01)
$n = 5 \cdot 10^5$, $k = 2500$	6.99 ($\pm$0.13)	4.90 ($\pm$0.02)
$n = 5 \cdot 10^6$, $k = 3000$	7.77 ($\pm$0.11)	5.48 ($\pm$0.01)
$n = 5 \cdot 10^6$, $k = 8000$	7.44 ($\pm$0.14)	5.41 ($\pm$0.01)
$n = 5 \cdot 10^7$, $k = 5000$	7.95 ($\pm$0.20)	5.84 ($\pm$0.02)

consists of 1000 points sampled from a uniform distribution on the hypersphere $\mathbb{S}^2 \subseteq \mathbb{R}^3$. We can see that $\hat{m}_k$ performs best and yields a perfect estimate for all values of k in the range of consideration. Our estimators perform well and yield a correct result after rounding for a broad range of k too. The right plot deals with 2000 points from a 7-dimensional Gaussian $N_7(0, I)$. In this higher-dimensional case the situation is different: while $E_{\text{CAP}}(V)$ still performs reasonably and yields a correct result, at least for a not too small range of k, $E_{\text{DP}}(V)$ constantly underestimates the dimension. This confirms our findings of Section 3.2.3.

Dependence on the sample size n

As we have proved in Section 3.3, both E_{CAP} and E_{DP} converge to the true dimension of the data set as $n \to \infty$ if k is chosen appropriately. In Table 3.2 we show the estimates from $E_{\text{CAP}}(R)$ and $E_{\text{DP}}(R)$ for increasing sample size n in the case of data points that are sampled from a 7-dimensional Gaussian $N_7(0, I)$. Here R denotes a random subset of V containing 10 vertices chosen uniformly at random without replacement. We can see that for E_{DP} the convergence is painfully slow. Even for $n = 5 \cdot 10^7$ sample points it still underestimates the dimension. E_{CAP} needs a lot fewer sample points in order to give a valuable result as we have seen already in the previous experiment. However, in Table 3.2 E_{CAP} has a tendency to slightly "overshoot", which is a consequence of a suboptimal choice of k.

Bias with respect to the true dimension

The left plot of Figure 3.5 shows the estimates from the estimators $\hat{m}_k$, $E_{\text{DP}}(V)$, and $E_{\text{CAP}}(V)$ as a function of the true dimension d when the data set consists of 5000 points from a d-dimensional Gaussian $N_d(0, I)$. The parameter k was set to 20. We can see that as the true dimension d increases, the property of underestimating the dimension of E_{DP} is shared by E_{CAP} and even by $\hat{m}_k$ (although to a much slighter extent).

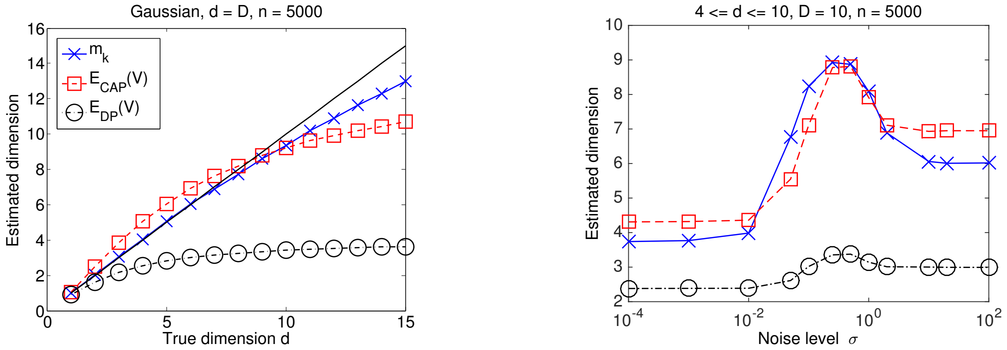


Figure 3.5: Left: The estimates from $\hat{m}_k$, $E_{\mathrm{DP}}(V)$, and $E_{\mathrm{CAP}}(V)$ as a function of the true dimension d (solid black line) for 5000 points from a d-dim Gaussian $N_d(0, I)$. Right: The estimates as a function of the noise level σ for 5000 points from $U(0,1)^4 \times N_6(0, \sigma I)$. In both experiments k was set to 20.

Noisy data

In the right plot of Figure 3.5 we can see the estimated dimensions as a function of the noise level σ for 5000 points drawn from $U(0,1)^4 \times N_6(0, \sigma I)$. Here $U(0,1)$ denotes a uniform distribution on the unit interval. As in the previous experiment, k was set to 20. When σ is small, the last six components of the data points can be considered as noise and the dimension of the data set should be four. As σ increases, the noise level gets so high that the data set actually should be considered as 10-dimensional. Finally, the role of the “true data” and the noise gets inverted, the first four components of the data points are dominated by the six last ones and considered as noise, hence the dimension should be 6. This behavior is reflected by the estimates from all three estimators under consideration. However, again the performance of $E_{\mathrm{DP}}(V)$ is very poor, completely failing to correctly determine either of the various dimensions.

Variance depending on $|A|$

Finally, we study the variance of our estimators depending on the size of A. In this experiment we consider data sets consisting of 1000 points drawn from a uniform distribution on the hypersphere $\mathbb{S}^2 \subseteq \mathbb{R}^3$, but similar observations hold for other data sets as well. We set $k = 15$. Figure 3.6 shows box plots of 10000 realizations of $E_{\mathrm{DP}}(A)$ (1st plot) or $E_{\mathrm{CAP}}(A)$ (2nd plot) with $|A| = 10$, $|A| = 50$, or $|A| = 1000$ (i.e., $A = V$). For every realization we created a new data set and A was chosen uniformly at random without replacement (when $|A| = 10$ or $|A| = 50$). We see that the estimates are the more concentrated the larger the size of A as expected. The third plot of Figure 3.6 shows the squared coefficient of variation CV^2 of 10000 realizations of $E_{\mathrm{DP}}(A)$ and $E_{\mathrm{CAP}}(A)$, respectively, as a function of $|A|$. The squared coefficient of variation is the ratio of the variance to the squared mean. For the whole range of $|A|$, CV^2 is smaller for $E_{\mathrm{DP}}(A)$ than for $E_{\mathrm{CAP}}(A)$. However, recall that $E_{\mathrm{DP}}(A)$ has a larger bias (this can also be seen from the box plots). The variance of $E_{\mathrm{CAP}}(A)$ decreases almost like $1/|A|$ over the whole range of $|A|$ as if the local statistics $L_{\mathrm{CAP}}(i)$ were independent among $i \in V$. For small values of $|A|$ this may be expected since then it is likely that most sample points x_i

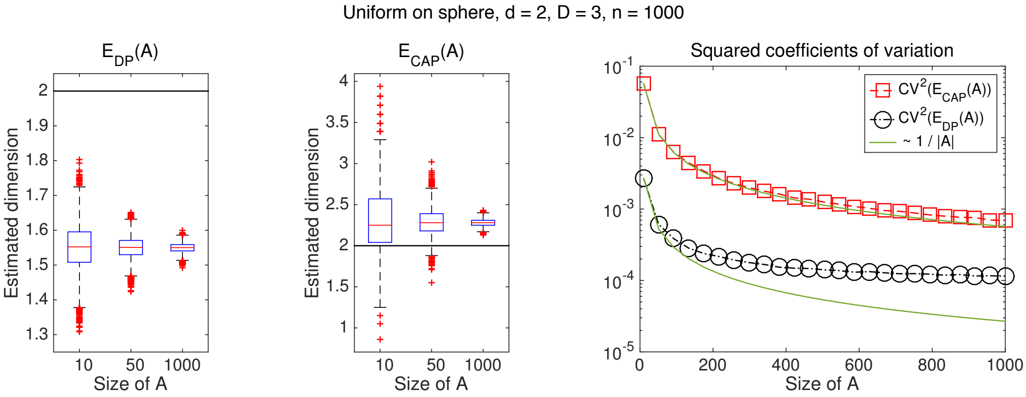


Figure 3.6: 1st & 2nd plot: Box plots of 10000 realizations of $E_{\mathrm{DP}}(A)$ and $E_{\mathrm{CAP}}(A)$, respectively, for various values of $|A|$. 3rd plot: The squared coefficient of variation of 10000 realizations of $E_{\mathrm{DP}}(A)$ and $E_{\mathrm{CAP}}(A)$, respectively, as a function of $|A|$. The green curves are proportional to $1/|A|$. The considered data sets consist of 1000 points from a uniform distribution on the hypersphere $\mathbb{S}^2 \subseteq \mathbb{R}^3$, and we set $k = 15$.

and x_j with $i, j \in A$ are far apart, in which case $L_{\mathrm{CAP}}(i)$ and $L_{\mathrm{CAP}}(j)$ are effectively independent. The same argument is true for $E_{\mathrm{DP}}(A)$, but compared to $L_{\mathrm{CAP}}(i)$ and $L_{\mathrm{CAP}}(j)$, x_i and x_j have to be further apart in order that $L_{\mathrm{DP}}(i)$ and $L_{\mathrm{DP}}(j)$ are effectively independent since these statistics are based on larger neighborhoods. For this reason the variance of $E_{\mathrm{CAP}}(A)$ decreases like $1/|A|$ only in a range of very small values of $|A|$, but does not significantly decline anymore in a range of large values.

3.5 Discussion

In this chapter we have introduced two estimators for the intrinsic dimension of a data set that require only the directed, but unweighted k-nearest neighbor graph on the data set as input. In fact, for producing an estimate of the intrinsic dimension locally around a data point x_i we do not need the whole graph, but knowledge of the k nearest neighbors of x_i and these neighbors' neighbors is sufficient. Although dimensionality estimation is a well-studied problem with a long history, all existing methods require cardinal distance information about the data set as input. Our estimators E_{DP} and E_{CAP} are the first ones that are based on only ordinal data. Under some mild regularity assumptions, we have proved that both our estimators are statistically consistent, that is they converge in probability to the true dimension as the number of data points, which are assumed to be sampled from some probability space, tends to infinity. In combination with our results of Chapter 2 our consistency result allows us to claim that abundant ordinal data of the type (1.1) about a Euclidean data set asymptotically contains all cardinal distance information up to rescaling, even when the intrinsic dimension of the data set is unknown. We provided theoretical evidence that our estimator E_{CAP} might be superior to our estimator E_{DP}, and this finding has been clearly confirmed by our experiments.

There is an obvious follow-up question to our work. It is similar to one of the questions raised in Section 2.5 when we discussed our results about the asymptotic uniqueness of ordinal embeddings: do we really need knowledge of the k nearest neighbors of some of the data points to be able to estimate the dimension of a data set, or would other types of ordinal data (with less information content—compare with Section 1.3) suffice as well? More generally, which types of ordinal data require which number of corresponding ordinal constraints such that we can come up with a consistent estimator based on only these constraints? Providing a strategy for reliably estimating the intrinsic dimension of a data set based on whatever ordinal distance information is available would be valuable for working with ordinal data in practice: when following an ordinal embedding approach (compare with Section 1.4) one has to choose the dimension of the space of the embedding, and one cannot expect the ordinal embedding to accurately represent a data set if the embedding dimension is chosen smaller than the intrinsic dimension of the data set. If the data points do not lie on a manifold, but rather "fill up" one or more regions of a Euclidean space, the intrinsic dimension might even be the optimal choice for the embedding dimension. Note that existing strategies for choosing the dimension of the space of an ordinal embedding are only heuristic. They consist of computing ordinal embeddings for various dimensions d. Then one can consider the curve that shows the values of the stress function at the computed embeddings over d and the final dimension is chosen by looking for an elbow in this curve (Kruskal, 1964a). Alternatively, one can choose the dimension of the ordinal embedding that best reflects a part of the ordinal data that was kept as a validation set (van der Maaten and Weinberger, 2012).

Chapter 4

Lens depth function and k-relative neighborhood graph: versatile tools for ordinal data analysis

Up to now, the main approach to solve a machine learning problem in a setting of ordinal distance information is to construct an ordinal embedding of the data set (compare with Section 1.4) and to solve the problem on the embedding using an algorithm for vector-valued data. However, such a two-step approach comes with a number of drawbacks in practice:

- All existing methods for ordinal embedding are based on numerical optimization. Their objective functions are not convex with respect to the points of the embedding, and hence optimizing with respect to the points directly involves the risk of finding only a suboptimal solution. In this case, the optimization process usually starts from a random initialization of the embedding and the ordinal embedding algorithm is non-deterministic. Some objective functions (e.g., the ones in the algorithms by Agarwal et al., 2007, Shaw and Jebara, 2009, or in one of the two algorithms proposed by van der Maaten and Weinberger, 2012) can be rephrased as convex functions of the Gram matrix of the embedding. This usually leads to a semidefinite program with trace regularization for the Gram matrix, from which the ordinal embedding is obtained via a singular value decomposition. Trace regularization is used as an approximation for the matrix rank in order to obtain a low-rank Gram matrix, and this again involves the risk of finding only a suboptimal solution.

- Regardless whether optimizing with respect to the points or the Gram matrix of the ordinal embedding, the considered optimization problems are expensive. For none of the existing ordinal embedding algorithms theoretical bounds for their complexity are available in the literature, but it is widely known that they are rather slow and not appropriate when dealing with large data sets and/or many ordinal constraints. We will also see this in the experiments of this chapter in Section 4.5.

- All ordinal embedding algorithms require to set parameters (besides the usual parameters required for numerical optimization): Most importantly, one has to choose the dimension of the space of the embedding—or a weight factor for the trace regularization of the Gram matrix that governs the dimension. The existing strategies for choosing the embedding dimension come without any theoretical guarantees (compare with Section 3.5). A bad choice might have severe implications, but this has not been seriously studied yet. Some algorithms (e.g., the one by Tamuz et al., 2011) require to choose additional model parameters, for which the consequences of a bad choice are not known either.

All these are strong arguments for aiming to solve machine learning problems in a setting of ordinal distance information directly, that is without constructing an ordinal embedding as an intermediate step. This is also in accordance with Vapnik's *main principle for solving problems using a restricted amount of information*, which says: "when solving a given problem, try to avoid solving a more general problem as an intermediate step" (Vapnik, 1998, Section 1.9).

In this chapter we propose algorithms for the problems of medoid estimation, outlier identification, classification, and clustering when given only a collection of statements of the kind ($\star$) (compare with Section 1.3) for a data set. Our algorithms do not construct an ordinal embedding as an intermediate step. They rather use the ordinal data to estimate the lens depth function and the k-relative neighborhood graph on the data set. These objects come from multivariate statistics and computer vision, respectively, and have been successfully used in machine learning before. Our algorithms are simple, are much faster than an ordinal embedding approach and avoid some of its other drawbacks, and can easily and highly efficiently be parallelized.

4.1 Setup for Chapter 4 and a closer look at statements of the kind ($\star$)

We assume to be given an arbitrary collection $\mathcal{S}$ of statements of the kind ($\star$) for some data set $\mathcal{D}$. Recall from Section 1.3 that a statement of the kind ($\star$) reads as

$$\textit{Object } A \textit{ is the most central object within the triple of objects } (A,B,C), \tag{$\star$}$$

where (A,B,C) is a triple of pairwise distinct objects in $\mathcal{D}$, and that this means that

$$\big(\iota(A,B) < \iota(B,C)\big) \;\;\wedge\;\; \big(\iota(A,C) < \iota(B,C)\big). \tag{4.1}$$

For simplicity, in this chapter we assume that there are no ties in the total order of all dissimilarities between distinct objects in $\mathcal{D}$, and hence there is always a unique most central object within a triple of objects. From (4.1) we see that the most central data point within a triple of data points is the data point opposite to the longest side in the triangle spanned by the three data points. An illustration of this can be seen on the left side of Figure 4.1. Since (4.1) is equivalent to

$$\big(\iota(A,B)+\iota(A,C)\big) < \big(\iota(B,A)+\iota(B,C)\big) \;\wedge\; \big(\iota(A,B)+\iota(A,C)\big) < \big(\iota(C,A)+\iota(C,B)\big),$$

object A being the most central object within (A,B,C) is equivalent to A being the medoid of $\{A,B,C\}$ (see Section 4.2.1 if you want to recall the definition of a medoid).

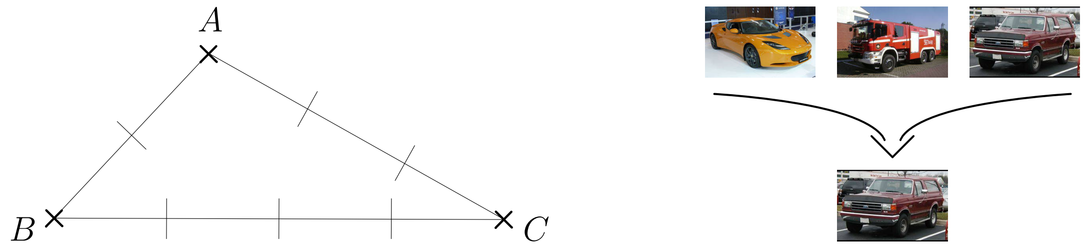


Figure 4.1: Illustration of the meaning of statement $(\star)$. Left: We have $\iota(A, B) < \iota(A, C) < \iota(B, C)$, and hence A is the most central data point within (A, B, C). Right: Within the three cars shown at the top, the off-road vehicle shown at the bottom a second time is the most central or best representative one.[1]

Intuitively, we can find the most central object within a triple by looking for the best representative of the triple. This allows us to easily collect statements of the kind $(\star)$ via crowdsourcing. Consider the example of a triple of cars consisting of a sports car, a fire truck, and an off-road vehicle shown on the right side of Figure 4.1: Obviously, the sports car and the fire truck are rather different, but the off-road vehicle is not so different from either of them. Hence, the off-road vehicle can most likely be taken for a representative of the three cars and is the most central object within the triple.

At this point we do not make any assumptions on how $\mathcal{S}$ is related to the set of all statements, that is the set of statements informing about the most central object within every possible triple (e.g., sampled uniformly at random). However, we need to make some assumptions if we want to provide a theoretical justification for our proposed algorithms (compare with Section 4.2). Statements might be repeatedly present in $\mathcal{S}$. More importantly, we allow $\mathcal{S}$ to be noisy and inconsistent (compare with Section 1.3).

4.2 Lens depth function and k-relative neighborhood graph and motivation for our algorithms

Ordinal distance information of the kind $(\star)$ has the attractive property that it is intimately related to the lens depth function and the k-relative neighborhood graph. In this section we introduce the lens depth function and the k-relative neighborhood graph. We discuss the relationship to ordinal data of the kind $(\star)$ and explain how we can exploit this relationship in order to devise algorithms for medoid estimation, outlier identification, classification, and clustering that only require a collection of statements $(\star)$ as input. This section only serves for providing intuitions and motivations. We will formally present our algorithms in the subsequent Section 4.3 and provide further background on the lens depth function and the k-relative neighborhood graph as well as references in Section 4.4.

The most important geometric object in the following is the lens spanned by two points $x_i, x_j \in \mathcal{X}$. Consider an open ball of radius $\iota(x_i, x_j)$ centered at x_i, and similarly

[1] The pictures of the cars were found on Wikimedia Commons and have been explicitly released into the public domain by their authors.

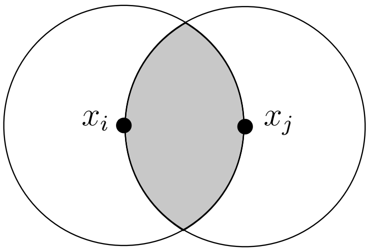


Figure 4.2: Illustration of $Lens(x_i, x_j)$ in case of the Euclidean plane. The lens is shown in grey.

an open ball of the same radius centered at x_j. The lens spanned by x_i and x_j consists of all those points of $\mathcal{X}$ that are located in the intersection of these two balls. Formally,

$$\begin{aligned} Lens(x_i, x_j) &= \{x \in \mathcal{X} : \iota(x, x_i) < \iota(x_i, x_j)\} \cap \{x \in \mathcal{X} : \iota(x, x_j) < \iota(x_i, x_j)\} \\ &= \big\{x \in \mathcal{X} : \max\{\iota(x, x_i), \iota(x, x_j)\} < \iota(x_i, x_j)\big\}. \end{aligned}$$

An illustration of $Lens(x_i, x_j)$ in case of the Euclidean plane, that is $\mathcal{X} = \mathbb{R}^2$ and ι equaling the Euclidean metric, can be seen in Figure 4.2. The key insight for us are the following equivalences:

$$\begin{aligned} x \in Lens(x_i, x_j) \Leftrightarrow \\ \iota(x, x_i) < \iota(x_i, x_j) \text{ and } \iota(x, x_j) < \iota(x_i, x_j) \Leftrightarrow \\ x \text{ is the most central point within } (x, x_i, x_j) \end{aligned} \tag{4.2}$$

In particular, if we had knowledge of all ordinal relationships of type ($\star$) for a data set $\mathcal{D} \subseteq \mathcal{X}$, we could check for any data point x_k and any two data points x_i, x_j whether x_k is contained in $Lens(x_i, x_j)$ or not.

4.2.1 Lens depth function

The lens depth function (Liu and Modarres, 2011) is an instance of a statistical depth function. These functions are a widely known tool in multivariate statistics. They have been designed to measure centrality with respect to point clouds or probability distributions. We will provide more information about statistical depth functions in general, including references, in Section 4.4.2.

What makes the lens depth function special for us is that it does not rely on Euclidean structures or numeric distance values. This is in contrast to all other depth functions from the literature. Given a data set $\mathcal{D} = \{x_1, \ldots, x_n\} \subseteq \mathcal{X}$, the lens depth function $LD(\,\cdot\,; \mathcal{D}) : \mathcal{X} \rightarrow \mathbb{N}_0$ is defined as

$$LD(x; \mathcal{D}) = \big|\, \{(x_i, x_j) : x_i, x_j \in \mathcal{D}, i < j, x \in Lens(x_i, x_j)\} \,\big|.$$

To understand its meaning, consider a set of data points in the Euclidean plane. A point located at the "heart of the set" will lie in the lenses of many pairs of data points. Thus the lens depth function will attain a high value at this point, indicating its high centrality. In contrast, points at the boundary of the point cloud will lie in only a few lenses

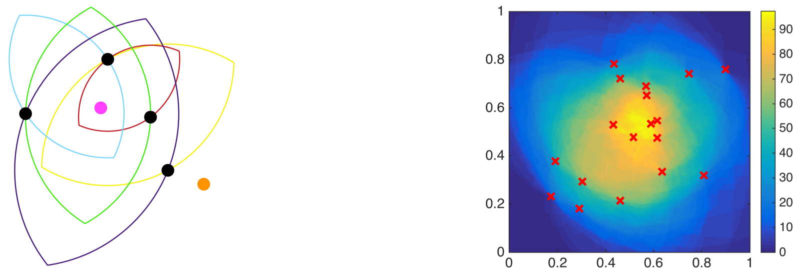


Figure 4.3: Left: The pink point at the center is contained in almost every lens spanned by any of two data points while the orange one located at the bottom right edge of the point set is not contained in a single lens. Right: Heat map of the lens depth function for a data set of 18 points (in red) in the unit square of the Euclidean plane.

and will have a low lens depth value, indicating their low centrality. See the left side of Figure 4.3 for an illustration. The right side of Figure 4.3 shows a heat map of the lens depth function for a data set consisting of 18 points in the Euclidean plane.

Making use of (4.2) we can see immediately how easily the lens depth function can be evaluated based on statements of the kind ($\star$). Given *all* statements of the kind ($\star$) for a data set $\mathcal{D} = \{x_1, \ldots, x_n\}$, that is one statement for every unordered triple (x_i, x_j, x_k) of distinct objects in $\mathcal{D}$, we can immediately *evaluate* $LD(x_t; \mathcal{D})$ for any $t \in \{1, \ldots, n\}$. It simply holds that

$$LD(x_t; \mathcal{D}) = \text{number of statements comprising } x_t \text{ as most central data point.} \tag{4.3}$$

We note that $LD(x_t; \mathcal{D})$ as given in (4.3) can be considered, up to a normalizing constant of $1/\binom{n-1}{2}$, as probability of the fixed data point x_t being the most central data point in a triple comprising x_t and two data points drawn uniformly at random without replacement from $\mathcal{D} \setminus \{x_t\}$. This insight gives us a handle for the realistic situation that we are not given all statements of the kind ($\star$), but only an arbitrary collection $\mathcal{S}$ of statements, some of them possibly being incorrect. Namely, we can still *estimate* $LD(x_t; \mathcal{D})$ by estimating the probability of the described event by its relative frequency:

$$LD(x_t; \mathcal{D}) \mathrel{\dot{\approx}} \frac{\text{number of statements in } \mathcal{S} \text{ that comprise } x_t \text{ as most central data point}}{\text{number of statements in } \mathcal{S} \text{ that comprise } x_t} \tag{4.4}$$

This estimate will be reasonable whenever statements in $\mathcal{S}$ comprising x_t appear to be sampled approximately uniformly at random from the set of all statements that comprise x_t, the number of statements in $\mathcal{S}$ comprising x_t is large enough, and the proportion of incorrect statements is sufficiently small. Note that if we assume $\mathcal{S}$ to be sampled uniformly at random from the set of all statements, this will imply that for every $x_t \in \mathcal{D}$ statements in $\mathcal{S}$ comprising x_t are a uniform sample from the set of all statements that comprise x_t.

We now want to explain how we can use our insights to devise algorithms for the machine learning problems of medoid estimation, outlier identification, and classification when only given a collection of statements of the kind ($\star$) for a data set (the algorithms are formally stated in Section 4.3). The basic principle is that we replace the true lens depth function with its estimate according to (4.4) in the following existing approaches to these problems (see Section 4.4.2 for further information and references):

- **Medoid estimation (cf. Algorithm 1 in Section 4.3):** A medoid O_{MED} of a data set $\mathcal{D}$ is a most central object in the sense that it has minimal total distance to all other objects, that is it minimizes
$$I(O) = \sum_{O_i \in \mathcal{D}} \iota(O, O_i), \quad O \in \mathcal{D}. \tag{4.5}$$
Since the lens depth function provides a measure of centrality too, even though in a different sense, a maximizer of the lens depth function (restricted to $\mathcal{D}$) is a natural candidate for an estimate of a medoid.

- **Outlier identification (cf. Algorithm 2 in Section 4.3):** An outlier in a data set $\mathcal{D}$ is "an observation . . . which appears to be inconsistent with the remainder of that set of data" (Barnett and Lewis, 1978, Chapter 1). Points with a low lens depth value are non-central points according to the lens depth function and thus are natural candidates for outliers. We will see in the experiments in Section 4.5.1 that this approach works well for data sets with a uni-modal structure, but can fail in multi-modal cases.

- **Classification (cf. Algorithm 3 in Section 4.3):** The simplest approach to classification based on the lens depth function is to assign a test point to that class in which it is a more central point: For each of the classes we can compute a separate lens depth function and evaluate a test point's corresponding depth value. The test point is then classified as belonging to the class that gives rise to the highest lens depth value. However, it has been found that such a *max-depth* approach has some severe limitations (compare with Section 4.4.2).

  To overcome these limitations, we use a feature-based approach. We consider the data-dependent feature map
$$x \mapsto (LD(x; Class_1), LD(x; Class_2), \ldots, LD(x; Class_K)) \in \mathbb{R}^K, \quad x \in \mathcal{X}, \tag{4.6}$$
and then apply an out-of-the-box classification algorithm to the K-dimensional representation of the data set.

4.2.2 k-relative neighborhood graph

We now use the lenses spanned by two data points in order to define the k-relative neighborhood graph (k-RNG). In our language, for a data set $\mathcal{D} = \{x_1, \ldots, x_n\} \subseteq \mathcal{X}$ and a parameter $k \in \mathbb{N}$ the k-RNG on $\mathcal{D}$ is the graph with vertex set $\mathcal{D}$ in which two distinct vertices x_i and x_j are connected by an undirected edge if and only if the lens spanned by these points contains fewer than k data points from $\mathcal{D}$:

$$x_i \sim x_j \;\Leftrightarrow\; |Lens(x_i, x_j) \cap \mathcal{D}| < k \tag{4.7}$$

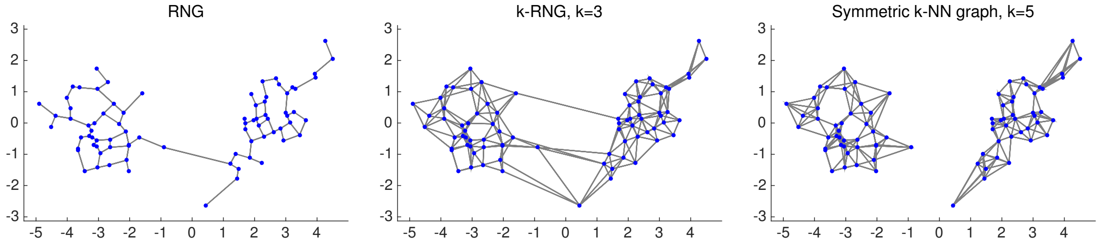


Figure 4.4: The RNG (left), the 3-RNG (middle), and the symmetric 5-NN graph (right) on 80 points from a mixture of two Gaussians in the Euclidean plane. Note that as opposed to k-NN graphs, k-relative neighborhood graphs tend to have more connections between points from the different mixture components. In fact, a k-RNG is always connected (see Section 4.4.3). This may be desirable in some situations, but undesirable in others.

The rationale behind this definition is that two data points may be considered close to each other whenever the lens spanned by them contains only a few data points. The k-relative neighborhood graph is best known when $k = 1$. In this form it is simply called relative neighborhood graph (RNG) and has already been introduced in Toussaint (1980). The general k-RNG seems to appear in Chang et al. (1992) for the first time. Examples for a data set in the Euclidean plane can be seen in Figure 4.4. For comparison, we also show the symmetric k-NN graph (for $k = 5$), which is more popular in machine learning. In that graph two vertices are connected by an undirected edge whenever one of them is among the k closest data points to the other one (with respect to the distance function ι).

Given *all* statements of the kind ($\star$) for a data set $\mathcal{D}$, it is straightforward to build the *true* k-RNG on $\mathcal{D}$ similarly to the exact evaluation of the lens depth function (4.3). We will discuss how to build an *estimate* of the k-RNG on $\mathcal{D}$ when given only an arbitrary collection of statements, some of them possibly being incorrect, and a problem involved shortly. Before, let us explain how k-relative neighborhood graphs can be used for classification and clustering. Again, these ideas are not new, but have already been discussed in the literature (see Section 4.4.3). As with the true lens depth function, in our algorithms presented in Section 4.3 we simply replace the true k-RNG with its estimate.

- **Classification (cf. Algorithm 4 in Section 4.3):** Given a set of labeled points and an additional test point that we want to classify, we can construct the k-RNG on the union of the set of labeled points and the singleton of the test point and take a majority vote of the test point's neighbors in the graph. Actually, there is no need to construct the whole graph. We just have to find the test point's neighbors in the graph. Note that the basic principle is the same as for the well-known k-NN classifier (e.g., Shalev-Shwartz and Ben-David, 2014, Chapter 19), replacing the directed k-NN graph by the k-RNG.

- **Clustering (cf. Algorithm 5 in Section 4.3):** As we can do with the symmetric k-NN graph, it is straightforward to apply spectral clustering (see, e.g., von Luxburg, 2007, for a comprehensive introduction) to the k-RNG on a data set $\mathcal{D}$. We propose two versions: one is to simply work with an estimate of the ordinary unweighted

k-RNG, the other one is to use an estimate of a k-RNG in which an edge between connected vertices x_i and x_j is weighted by

$$\exp\left(-\frac{1}{\sigma^2}\cdot\frac{|Lens(x_i,x_j)\cap\mathcal{D}|^2}{(|\mathcal{D}|-2)^2}\right) \tag{4.8}$$

for a scaling parameter $\sigma > 0$.

The problem of estimating the k-RNG from noisy ordinal data

The key insight for estimating the k-RNG on a data set $\mathcal{D}$ based on ordinal distance information of type ($\star$) is similar to the one for estimating the lens depth function: the characterization (4.7) is equivalent to two distinct, fixed data points x_i and x_j being connected in the k-RNG if and only if the probability of a data point drawn uniformly at random from $\mathcal{D}\setminus\{x_i,x_j\}$ lying in $Lens(x_i,x_j)$ is smaller than $k/(|\mathcal{D}|-2)$. Given a collection $\mathcal{S}$ of statements of the kind ($\star$), this probability can be estimated by

$$V(x_i,x_j)=\frac{N(x_i,x_j)}{D(x_i,x_j)} \tag{4.9}$$

where

$$\begin{aligned} N(x_i,x_j) &= \text{number of statements in } \mathcal{S} \text{ comprising both } x_i \text{ and } x_j \text{ and} \\ &\quad\ \text{another data point as most central data point,} \\ D(x_i,x_j) &= \text{number of statements in } \mathcal{S} \text{ comprising both } x_i \text{ and } x_j. \end{aligned} \tag{4.10}$$

Thus our strategy to estimate the k-RNG on $\mathcal{D}$ is the following: we connect two data points x_i and x_j with $i\neq j$ by an undirected edge if and only if

$$V(x_i,x_j)<\frac{k}{|\mathcal{D}|-2}. \tag{4.11}$$

If all statements in $\mathcal{S}$ are correct and, for every x_i and x_j with $i\neq j$, there are sufficiently many statements in $\mathcal{S}$ that comprise both x_i and x_j and these statements appear to be sampled approximately uniformly at random from the set of all statements that comprise x_i and x_j, we can expect our estimate of the k-RNG to be reasonable.

However, incorrect statements in $\mathcal{S}$ create a problem for our strategy. Usually, we are interested in a k-RNG for a small value of the parameter k, aiming at connecting only data points that are close to each other. Consequently, according to (4.11), in order that the data points x_i and x_j are connected in our estimate of the k-RNG, the estimated probability $V(x_i,x_j)$ has to be small. However, in case of erroneous ordinal data comprising sufficiently many incorrect statements, there will always be statements wrongly indicating that there are some data points in $Lens(x_i,x_j)$ that in fact are not, and thus $V(x_i,x_j)$ will always be somewhat large. Hence, many of the edges of the true k-RNG on $\mathcal{D}$ will not be present in our estimate.

To make this formal, consider the following simple noise model: Statements of the kind ($\star$) are incorrect, independently of each other, with some fixed probability *errorprob*. In an incorrect statement the two data points that in fact are not most central appear to be most central with probability 1/2 each. Assume $\mathcal{S}$ to be sampled uniformly at random from all statements. Denote by $p = p(x_i, x_j)$ the probability that a data point drawn uniformly at random from $\mathcal{D} \setminus \{x_i, x_j\}$ lies in $Lens(x_i, x_j)$, that is $p = |Lens(x_i, x_j) \cap \mathcal{D}|/(|\mathcal{D}| - 2)$. Denote by $\tilde{p} = \tilde{p}(x_i, x_j)$ the probability that the following experiment yields a positive result: A data point is drawn uniformly at random from $\mathcal{D} \setminus \{x_i, x_j\}$. Independently, a Bernoulli trial with a probability of success equaling *errorprob* is performed. If the Bernoulli trial fails, the experiment yields a positive result if and only if the drawn data point falls into $Lens(x_i, x_j)$. If the Bernoulli trial succeeds, the experiment yields a positive result if and only if the data point does not fall into $Lens(x_i, x_j)$ and another Bernoulli trial, with a probability of success of one half and performed independently, succeeds. It is clear that under the considered model, $V(x_i, x_j)$ as given in (4.9) and (4.10) is an estimate of $\tilde{p}$ rather than of p. Assuming that *errorprob* is less than 2/3, we can relate $\tilde{p}$ and p via

$$\tilde{p} = p \cdot (1 - errorprob) + (1 - p) \cdot errorprob \cdot \frac{1}{2}, \tag{4.12}$$

or equivalently

$$p = \frac{\tilde{p} - \frac{1}{2} \cdot errorprob}{1 - \frac{3}{2} \cdot errorprob}. \tag{4.13}$$

The probability $\tilde{p}$ is obtained from p by applying an affine transformation and vice versa. It follows from (4.12) that our strategy yields an estimate of the k'-RNG with

$$k' = \frac{k - \frac{1}{2} \cdot errorprob \cdot (|\mathcal{D}| - 2)}{1 - \frac{3}{2} \cdot errorprob} \tag{4.14}$$

rather than of the intended k-RNG. In particular, we have $k' < k$ for $k < \frac{1}{3}(|\mathcal{D}| - 2)$ and $k' \leq 0$ for $k \leq \frac{1}{2} \cdot errorprob \cdot (|\mathcal{D}| - 2)$. This means that whenever $k < \frac{1}{3}(|\mathcal{D}| - 2)$, our strategy produces an estimate containing fewer edges than we would like to have, and if $k \leq \frac{1}{2} \cdot errorprob \cdot (|\mathcal{D}| - 2)$, it even produces an estimate of an empty graph, that is a graph without any edges at all.

These findings might seem worse than they actually are: using our estimated graph for classification or clustering, we do not care whether we work with the estimate of a k'-RNG instead of a k-RNG, but only whether our classification or clustering result is useful. However, we have to bear them in mind when choosing the parameter k in our algorithms: Using cross-validation for choosing k for Algorithm 4 (classification by means of a majority vote of neighbors in the graph), we may only use Leave-one-out cross-validation variants since we have to ensure roughly the same size of the training set during cross-validation and the training set in the ultimate classification task. Otherwise, a value of k that is optimal during cross-validation will not be optimal in the ultimate classification problem since k' depends on $|\mathcal{D}|$ as stated in (4.14). Applying Algorithm 5 (spectral clustering on the estimated k-RNG), we have to choose k so large that the constructed graph is connected. This is not only required by some versions of spectral clustering, but also indicates that the graph is indeed an estimate of a true k'-RNG with

$k' \geq 1$ rather than of an empty graph. If we know the value of *errorprob*, or have at least an estimate of it, we can correct for the bias of our strategy. In order to estimate the k-RNG on a data set $\mathcal{D}$ for the intended value of k, according to (4.13), two data points x_i and x_j with $i \neq j$ should be connected if and only if

$$\frac{V(x_i, x_j) - \frac{1}{2} \cdot \mathit{errorprob}}{1 - \frac{3}{2} \cdot \mathit{errorprob}} < \frac{k}{|\mathcal{D}| - 2}, \tag{4.15}$$

which equals (4.11) if $\mathit{errorprob} = 0$. Note that although the left-hand side of equation (4.15) is an unbiased estimator of $p(x_i, x_j)$ for every x_i and x_j with $i \neq j$ (assuming $\mathcal{S}$ to be sampled uniformly at random from all statements), due to the thresholding step in (4.15) our estimation strategy is still not an unbiased estimator of the intended k-RNG.

4.3 Algorithms for medoid estimation, outlier identification, classification, and clustering

In this section we formally state our algorithms for the problems of medoid estimation, outlier identification, classification, and clustering when the only available information about a data set $\mathcal{D}$ is a collection $\mathcal{S}$ of statements of the kind ($\star$). Furthermore, we discuss running times, space requirements, and some implementation aspects.

4.3.1 Medoid estimation

The following Algorithm 1 returns as output an estimate of a medoid of $\mathcal{D}$ as motivated in Section 4.2.1. The estimate is given by an object that maximizes the estimated lens depth function on $\mathcal{D}$. By setting the estimated lens depth value $LD(O)$ to zero for objects O that do not appear in any statement in $\mathcal{S}$, which means that we do not have any information about O, we ensure that such an object is never returned as output.

Algorithm 1 Estimating a medoid

Input: a collection $\mathcal{S}$ of statements of the kind ($\star$) for some data set $\mathcal{D}$

Output: an estimate of a medoid of $\mathcal{D}$

```
1: for every object $O$ in $\mathcal{D}$ compute
       $$LD(O) := \frac{\text{number of statements comprising } O \text{ as most central object}}{\text{number of statements comprising } O}$$
   $\triangleright$ if the denominator equals zero, set $LD(O) = 0$
2: return an object $O$ for which $LD(O)$ is maximal
```

If we assume that every object in $\mathcal{D}$ can be identified by a unique index from $\{1, \ldots, |\mathcal{D}|\}$ and, given a statement in $\mathcal{S}$, the indices of the three objects involved can be accessed in constant time, then Algorithm 1 can be implemented with $\mathcal{O}(|\mathcal{D}| + |\mathcal{S}|)$ time and $\mathcal{O}(|\mathcal{D}|)$ space in addition to storing $\mathcal{S}$. This can be done by going through $\mathcal{S}$ only once and updating counters for the three objects found in a statement. If the objects in $\mathcal{D}$ are not indexed by $1, \dots, |\mathcal{D}|$, we can use minimal perfect hashing in order to first create such an indexing. This requires about $\mathcal{O}(|\mathcal{D}|)$ time and space (Hagerup and Tholey, 2001, Botelho et al., 2007), so the overall requirements remain unaffected by this additional step. An important feature of Algorithm 1 is that it can easily be parallelized by partitioning $\mathcal{S}$ into several subsets that may be processed independently. Since one usually may expect that $|\mathcal{S}| \gg |\mathcal{D}|$, such a parallelization has almost ideal speedup, that is doubling the number of processing elements leads to almost only half of the running time.

4.3.2 Identifying outlier candidates

By means of the following Algorithm 2 we can identify outliers in $\mathcal{D}$ given as input only a collection $\mathcal{S}$ of statements of the kind $(\star)$. Outlier candidates are data points with low estimated lens depth values $LD(O)$. By setting $LD(O)$ to zero for objects O that do not appear in any statement we guarantee that such objects are identified as outliers.

Algorithm 2 Identifying outlier candididates

Input: a collection $\mathcal{S}$ of statements of the kind $(\star)$ for some data set $\mathcal{D}$

Output: a subset of $\mathcal{D}$ containing objects that are outlier candidates

```
1: for every object $O$ in $\mathcal{D}$ compute
       $LD(O) := \frac{\text{number of statements comprising } O \text{ as most central object}}{\text{number of statements comprising } O}$
   $\triangleright$ if the denominator equals zero, set $LD(O) = 0$
2: identify objects with exceptionally small values of $LD(O)$
3: return the set of identified objects
```

The only difference between Algorithm 2 and Algorithm 1 is that instead of returning the object with the highest value of $LD(O)$ as estimate of a medoid we return objects with exceptionally small values as outlier candidates. The running time of Algorithm 2 depends on the identification strategy in Step 2, but if one simply identifies c objects with smallest values ($1 \leq c \leq |\mathcal{D}|$), then Algorithm 2 can be implemented with $\mathcal{O}(|\mathcal{D}| + |\mathcal{S}|)$ time and $\mathcal{O}(|\mathcal{D}|)$ space in addition to storing $\mathcal{S}$ analogously to Algorithm 1. Here we make use of the fact that the selection of the c-th smallest value in an array of length $|\mathcal{D}|$ can be done in $\mathcal{O}(|\mathcal{D}|)$ time and space (Blum et al., 1973). Just as for Algorithm 1, the first step of Algorithm 2 can easily be parallelized.

4.3.3 Classification

We propose two different algorithms for dealing with K-class classification in a data set $\mathcal{D}$ consisting of a subset $\mathcal{L}$ of labeled objects and a subset $\mathcal{U}$ of unlabeled objects when given no more information than the class labels for the objects in $\mathcal{L}$ and a collection $\mathcal{S}$ of statements of the kind $(\star)$ for $\mathcal{D}$. Our goal is to predict a class label for every object in $\mathcal{U}$.

Our first proposed algorithm, Algorithm 3, is based on the lens depth function and has been motivated in Section 4.2.1. It consists of computing a feature embedding of $\mathcal{D}$ into $[0,1]^K \subseteq \mathbb{R}^K$, in which each feature corresponds to the estimated lens depth value with respect to one class, and subsequently applying a classification algorithm that is suitable for K-class classification on $\mathbb{R}^K$ to this embedding.

Algorithm 3 K-class classification I

```
Input: a collection $\mathcal{S}$ of statements of the kind ($\star$) for some data set $\mathcal{D}$ comprising a
    set $\mathcal{L}$ of labeled objects and a set $\mathcal{U}$ of unlabeled objects; a class label for every
    labeled object in $\mathcal{L}$ according to its membership in one of $K$ classes (referred to as
    $Class_1, \ldots, Class_K$)
    ▷ note that we have $\mathcal{D} = \mathcal{L} \dot\cup \mathcal{U}$ and $\mathcal{L} = Class_1 \dot\cup Class_2 \dot\cup \ldots \dot\cup Class_K$
Output: an inferred class label for every unlabeled object in $\mathcal{U}$
1: for every object $O$ in $\mathcal{D}$ and $i \in \{1, \ldots, K\}$ compute
    $N_{C_i}(O) :=$ number of statements comprising $O$ and two labeled objects from $Class_i$
                  with $O$ as most central object
    $D_{C_i}(O) :=$ number of statements comprising $O$ and two labeled objects from $Class_i$
    $LD_{C_i}(O) := \frac{N_{C_i}(O)}{D_{C_i}(O)}$        ▷ if $D_{C_i}(O)$ equals zero, set $LD_{C_i}(O) = 0$
2: train an arbitrary classifier (suitable for $K$-class classification on $\mathbb{R}^K$) with training
    data
        $\{(LD_{C_1}(O_l), LD_{C_2}(O_l), \ldots, LD_{C_K}(O_l)) : O_l \in \mathcal{L}\} \subseteq \mathbb{R}^K,$
    where the label of $(LD_{C_1}(O_l), LD_{C_2}(O_l), \ldots, LD_{C_K}(O_l))$ equals the label of $O_l$
3: return as inferred class label of every unlabeled object $O_u \in \mathcal{U}$ the label predicted
    by the classifier applied to $(LD_{C_1}(O_u), LD_{C_2}(O_u), \ldots, LD_{C_K}(O_u)) \in \mathbb{R}^K$
```

Assuming that the number of classes K is bounded by a constant, the first step of Algorithm 3 requires $\mathcal{O}(|\mathcal{D}| + |\mathcal{S}|)$ operations and $\mathcal{O}(|\mathcal{D}|)$ space in addition to storing $\mathcal{S}$ and its implementation is similar to the one of Algorithm 1. As before, this step can easily and highly efficiently be parallelized (assuming that $|\mathcal{S}| \gg |\mathcal{D}|$). The time and space complexities of the remaining steps depend on the generic classifier that is used.

Our second proposed algorithm, Algorithm 4 (see page 89), is based on the k-RNG and has been motivated in Section 4.2.2. It is an instance-based learning method like the well-known k-NN classifier: There is no explicit training phase involved. An unlabeled object is readily classified by assigning the label that is most frequently encountered among the neighbors of the unlabeled object in the estimated k-RNG.

Assuming that the number of classes K is bounded by a constant, Algorithm 4 can be implemented with $\mathcal{O}(|\mathcal{D}|+|\mathcal{U}|\cdot|\mathcal{L}|+|\mathcal{S}|) = \mathcal{O}(|\mathcal{U}|\cdot|\mathcal{L}|+|\mathcal{S}|)$ time and $\mathcal{O}(|\mathcal{D}|+|\mathcal{U}|\cdot|\mathcal{L}|) = \mathcal{O}(|\mathcal{U}| \cdot |\mathcal{L}|)$ space in addition to storing $\mathcal{S}$. Here we have to assign to each labeled object

Algorithm 4 K-class classification II

```
Input: a collection $\mathcal{S}$ of statements of the kind $(\star)$ for some data set $\mathcal{D}$ comprising a
    set $\mathcal{L}$ of labeled objects and a set $\mathcal{U}$ of unlabeled objects; a class label for every
    labeled object in $\mathcal{L}$ according to its membership in one of $K$ classes (referred to as
    $Class_1, \ldots, Class_K$); an integer parameter $k$
    ▷ note that we have $\mathcal{D} = \mathcal{L} \mathbin{\dot{\cup}} \mathcal{U}$ and $\mathcal{L} = Class_1 \mathbin{\dot{\cup}} Class_2 \mathbin{\dot{\cup}} \ldots \mathbin{\dot{\cup}} Class_K$
Output: an inferred class label for every unlabeled object in $\mathcal{U}$

for every unlabeled object $O_u \in \mathcal{U}$ and every labeled object $O_l \in \mathcal{L}$ compute
    $N(O_u, O_l) :=$ number of statements comprising both $O_u$ and $O_l$ and another
                    labeled object as most central object
    $D(O_u, O_l) :=$ number of statements comprising both $O_u$ and $O_l$ and another
                    labeled object
    $V(O_u, O_l) := \frac{N(O_u, O_l)}{D(O_u, O_l)}$     ▷ if $D(O_u, O_l)$ equals zero, set $V(O_u, O_l) = \infty$
return as inferred class label of every unlabeled object $O_u \in \mathcal{U}$ the majority vote
    (ties broken randomly) of the labels of those objects $O_l \in \mathcal{L}$ that satisfy
        $V(O_u, O_l) < \frac{k}{|\mathcal{L}| - 1}$
```

a unique identifier in $\{1, \ldots, |\mathcal{L}|\}$ and to each unlabeled object a unique identifier in $\{1, \ldots, |\mathcal{U}|\}$ that can be looked up in constant time. This allows us to increment a value of $N(O_u, O_l)$ or $D(O_u, O_l)$ for $(O_u, O_l) \in \mathcal{U} \times \mathcal{L}$ stored in an array of size $|\mathcal{U}| \times |\mathcal{L}|$ within constant time. Once the objects are indexed by $1, \ldots, |\mathcal{D}|$ (compare with Section 4.3.1), we can easily assign such identifiers in $\mathcal{O}(|\mathcal{D}|)$ time and space. Again, it is straightforward to parallelize Algorithm 4 by partitioning $\mathcal{S}$.

4.3.4 Clustering

Our proposed Algorithm 5 (see page 90) for clustering a data set $\mathcal{D}$ when only given a collection $\mathcal{S}$ of statements of the kind $(\star)$ as input consists of estimating a k-RNG on $\mathcal{D}$ and applying spectral clustering to the estimate. Note that some versions of spectral clustering require the underlying similarity graph not to contain isolated vertices. A true k-RNG never contains isolated vertices since a k-RNG is always connected (compare with Section 4.4.3), but if k is chosen too small, an estimated k-RNG might contain isolated vertices (compare with Section 4.2.2).

The first step of Algorithm 5 can be implemented with $\mathcal{O}(|\mathcal{D}|^2 + |\mathcal{S}|)$ time and $\mathcal{O}(|\mathcal{D}|^2)$ space in addition to storing $\mathcal{S}$. It can be parallelized in the same way as the corresponding parts of the previous algorithms. However, here we achieve almost ideal speedup only in case $|\mathcal{S}| \gg |\mathcal{D}|^2$. The second step can be implemented with $\mathcal{O}(|\mathcal{D}|^2)$ time and $\mathcal{O}(|\mathcal{D}|^2)$ space. The complexity of Step 3 is the one of spectral clustering

Algorithm 5 Clustering

```
Input: a collection $\mathcal{S}$ of statements of the kind $(\star)$ for some data set $\mathcal{D} = \{O_1, \ldots, O_n\}$;
    an integer parameter $k$; number $l$ of clusters to construct; a parameter $\sigma > 0$ in case
    of weighted version
Output: a hard clustering $C_1, \ldots, C_l \subseteq \mathcal{D}$ with $C_1 \,\dot\cup\, C_2 \,\dot\cup\, \ldots \,\dot\cup\, C_l = \mathcal{D}$
1: for every pair $(O_i, O_j)$ of objects in $\mathcal{D}$ compute
       $N(O_i,O_j) :=$ number of statements comprising both $O_i$ and $O_j$
                      and another object as most central object
       $D(O_i,O_j) :=$ number of statements comprising both $O_i$ and $O_j$
       $V(O_i,O_j) := \frac{N(O_i,O_j)}{D(O_i,O_j)}$
   $\triangleright$ if $D(O_i,O_j) = 0$, set $V(O_i,O_j) = \infty$ (in particular, $V(O_i,O_i) = \infty$ for $i = 1, \ldots, n$)
2: let $W = (w_{ij})_{i,j=1,\ldots,n}$ be a $(n,n)$-matrix and either set
       $W_{ij} = \begin{cases} 1 & \text{if } V(O_i,O_j) < k/(|\mathcal{D}|-2) \\ 0 & \text{else} \end{cases}$      $\triangleright$ unweighted version
   or
       $W_{ij} = \begin{cases} e^{-\frac{V(O_i,O_j)^2}{\sigma^2}} & \text{if } V(O_i,O_j) < k/(|\mathcal{D}|-2) \\ 0 & \text{else} \end{cases}$      $\triangleright$ weighted version
3: apply spectral clustering to $W$ with $l$ as input parameter for the number of clusters
4: return clusters $C_1, \ldots, C_l$ according to the clusters produced in Step 3
```

after the construction of a similarity graph. Its costs are dominated by the complexity of eigenvector computations and are commonly stated to be in general in $\mathcal{O}(n^3) = \mathcal{O}(|\mathcal{D}|^3)$ regarding time and $\mathcal{O}(n^2) = \mathcal{O}(|\mathcal{D}|^2)$ regarding space for an arbitrary number of clusters l, unless approximations are applied (Yan et al., 2009, Li et al., 2011). In many cases the estimate of the k-RNG constructed by Algorithm 5 might be sparse (compare with Section 4.4.3), and then the eigenvector computations can be done much more efficiently (Bai et al., 2000). However, in the worst case the overall running time of Algorithm 5 can be up to $\mathcal{O}(|\mathcal{D}|^3 + |\mathcal{S}|)$. The overall space requirements are $\mathcal{O}(|\mathcal{D}|^2)$ in addition to storing $\mathcal{S}$.

4.4 Related work and further background

In this section we present related work and further background on statistical depth functions and k-relative neighborhood graphs.

4.4.1 Related work: algorithms based on ordinal data of type ($\boxplus$)

In the machine learning literature on ordinal distance information we are the first that consider statements of the kind ($\star$). However, these statements are closely related to statements of the kind ($\boxplus$) (compare with Section 1.3), which read as

$$\textit{Object } A \textit{ is the outlier within the triple of objects } (A, B, C). \tag{$\boxplus$}$$

In contrast to a statement of the kind ($\star$), which informs about the *most* central object within a triple of objects, a statement of the kind ($\boxplus$) informs about the *least* central object. Heikinheimo and Ukkonen (2013) have proposed strategies based on statements of the kind ($\boxplus$) that are very similar in spirit to our proposed Algorithms 1 and 2. Their algorithm for estimating a medoid of a data set works as follows: for every fixed data point they estimate the probability that the data point is the outlier within a triple of three data points containing the fixed data point and two data points chosen uniformly at random from the remaining ones and then return a data point with minimal estimated probability as estimate for a medoid. Similarly, they suggest to find outliers by looking for data points for which the estimated probability of being the outlier within a triple is exceptionally high. The only difference compared to our Algorithms 1 and 2 is that Heikinheimo and Ukkonen estimate the probability of being an outlier within a triple while we estimate the probability of being a most central object, and that they are looking for points with a low estimated probability when we are looking for points with a high estimated probability and vice versa. However, the conceptual problem with their approach is that the function that it is based on,

$$F(x; P) = 1 - \textit{Probability}(x \text{ is the outlier within the triple of points } (x, X, Y)), \tag{4.16}$$

where $x \in \mathcal{X}$, P is a probability distribution on $\mathcal{X}$, and X, Y are independent $\mathcal{X}$-valued random variables distributed according to P, is not a valid statistical depth function. It does not satisfy one of the most crucial properties of statistical depth functions, namely maximality at the center for symmetric distributions on $\mathbb{R}^d$ (see the following Section 4.4.2). As a consequence, there are data sets for which the approach by Heikinheimo and Ukkonen always fails to return a true medoid, even though given access to the correct statements of the kind ($\boxplus$) for all triples of data points. We will see in the experiments in Section 4.5.1 that our Algorithm 1 consistently achieves better results in recovering a true medoid of a data set compared to the method by Heikinheimo and Ukkonen.

4.4.2 Lens depth function and statistical depth functions in general

Statistical depth functions (see, e.g., Serfling, 2006, Cascos, 2009, Mosler, 2013, or the introduction of the dissertation of Van Bever, 2013, for basic reviews) have been developed to generalize the concept of the univariate median to multivariate distributions. To this end, a depth function is supposed to measure the centrality of all points $x \in \mathbb{R}^d$ with respect to a probability distribution, in the sense that the depth value at x is high if x resides in the "middle" of the distribution and that it is lower the more distant from the mass of the distribution x is located.

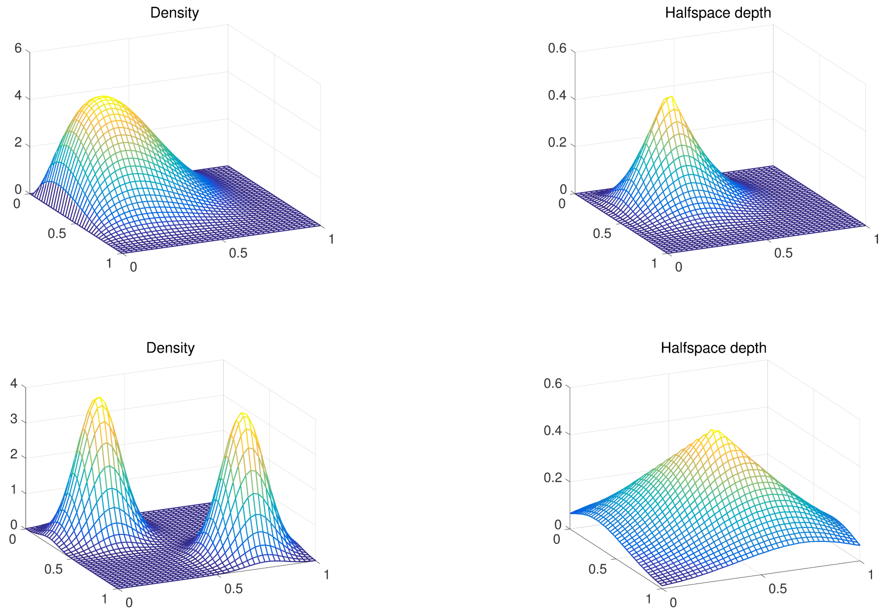


Figure 4.5: Illustration of the halfspace depth function. Mesh plots of the density and the halfspace depth function of a product of two $Beta(2,4)$-distributions (top row) and a mixture of two Gaussians (bottom row), respectively.

The first statistical depth function has been proposed by Tukey (1974). Given a probability distribution P on $\mathbb{R}^d$, the seminal halfspace depth function HD maps every point $x \in \mathbb{R}^d$ to the smallest probability of a closed halfspace containing x, that is

$$HD(x;P) = \inf_{u \in S^{d-1}} P(\{y \in \mathbb{R}^d : \langle u, y-x \rangle \geq 0\}),$$

where $S^{d-1} = \{u \in \mathbb{R}^d : \|u\| = 1\}$ denotes the unit sphere in $\mathbb{R}^d$. The intuition behind this definition is simplest to understand in case of an absolutely continuous distribution P: in this case $HD(x;P) \leq 1/2$, $x \in \mathbb{R}^d$, and in order for a point x to be considered central with respect to P it should hold that any hyperplane passing through x splits $\mathbb{R}^d$ into two halfspaces of almost equal probability $1/2$. Hence, points x are considered more central the higher their halfspace depth value $HD(x;P)$ is, and any point maximizing $HD(\,\cdot\,;P)$ is called a Tukey median. Figure 4.5 shows examples of the halfspace depth function for two absolutely continuous distributions on $\mathbb{R}^2$. Note that a depth function can resemble the density function of the underlying distribution only in case of a unimodal distribution—as a measure of global centrality depth functions are intended to be unimodal. We will take this up again in Section 4.5.1 and Section 4.6.

For a univariate and continuous distribution any ordinary median is also a Tukey median. In addition, the halfspace depth function HD satisfies a number of desirable properties:

1. Affine invariance: HD considered as a function in both x and P is invariant under affine transformations.

2. Maximality at the center: for a (halfspace) symmetric distribution the center of symmetry is a Tukey median.

3. Monotonicity with respect to the deepest point: if there is a unique Tukey median μ, $HD(x;P)$ decreases as x moves away along a ray from μ.

4. Vanishing at infinity: $HD(x;P) \to 0$ as $\|x\| \to \infty$.

Even though there is not a unique definition of a statistical depth function, these or closely related properties are typically requested for a function to qualify as depth function. Beside Tukey's halfspace depth, prominent examples of depth functions are simplicial depth (Liu, 1988, 1990), majority depth, projection depth, or Mahalanobis depth (Liu, 1992, Zuo and Serfling, 2000). To the best of our knowledge, the lens depth function (Liu and Modarres, 2011) is the only statistical depth function from the literature that can be evaluated given only ordinal distance information about a data set in an arbitrary set $\mathcal{X}$, which is equipped with an arbitrary dissimilarity function ι. Note once more that the function F defined in (4.16), which the approach by Heikinheimo and Ukkonen (2013) is based on, is provably not a statistical depth function. It does not satisfy the property of maximality at the center for symmetric distributions. Indeed, as Heikinheimo and Ukkonen observe, in case of a symmetric bimodal distribution in one dimension with the two modes sufficiently far apart, the center of symmetry is in fact a minimizer of F.

We provide some references related to our Algorithms 2 and 3: The idea of considering data points with a small depth value as outliers has been thoroughly studied in the setting of a contamination model in Chen et al. (2009) and Dang and Serfling (2010). In particular, they deal with the question of determining what a *small* depth value is.

The simple max-depth approach to binary classification outlined in Section 4.2.1 has already been proposed by Liu (1990), using simplicial depth instead of the lens depth function. It has been theoretically studied in Ghosh and Chaudhuri (2005). Ghosh and Chaudhuri proved that the max-depth approach is consistent, that is it asymptotically achieves Bayes risk, for equally probable and elliptically symmetric classes that only differ in location when using one of several depth functions and dealing with general K-class problems. Working not too well when these assumptions are not satisfied, the max-depth approach has been refined by Li et al. (2012) by allowing for more general classifiers on the DD-plot, thus overcoming some of its original limitations. The DD-plot (depth vs. depth plot; introduced by Liu et al., 1999) is the image of the data under the feature map $x \mapsto (DF(x;Class_1), DF(x;Class_2)) \in \mathbb{R}^2$, where DF denotes the depth function under consideration. Interestingly, Li et al. again only consider the 2-class case and propose a one-vs-one approach for the general case, which is different from our strategy of simply considering

$$x \mapsto (DF(x;Class_1), DF(x;Class_2), \ldots, DF(x;Class_K)) \in \mathbb{R}^K$$

as feature map and subsequently performing classification on $\mathbb{R}^K$.

We conclude this section with some comments about the lens depth function. An early version of the lens depth function has already been mentioned, but not seriously studied, by Lawrence (1996, Section 2.3) and by Bartoszynski et al. (1997). The main reference is Liu and Modarres (2011), where the lens depth function has been defined and systematically investigated. However, after reading the proofs in detail, we found that there is still an important gap. Liu and Modarres (2011) claim that the lens depth function satisfies the property of maximality at the center for centrally symmetric distributions on $\mathbb{R}^d$ (Theorem 6 in their paper). However, there is an error in their proof. It is *not* true that, conditioning on X_1, the probability of X_2 falling into a region such that $t \in Lens(X_1, X_2)$ holds decreases as $t \in \mathbb{R}^d$ moves away from the center for *all* values of X_1, and hence the monotonicity of the integral is not guaranteed. The same mistake appears in Elmore et al. (2006) and in Section 2.5 of Yang (2014) when showing the property for the spherical depth function and the β-skeleton depth function, respectively. So it has not yet been established that the lens depth function satisfies this essential property of statistical depth functions. We were not able to fix the proof, but we still believe that the statement is correct. At least, unlike for the function F defined in (4.16), we have not been able to construct any example of a symmetric distribution for which the lens depth function does not attain its maximum at the center.

4.4.3 k-relative neighborhood graph

The k-RNG belongs to the class of proximity graphs. In a graph belonging to this class two vertices are connected if they are in some sense close to each other (see Jaromczyk and Toussaint, 1992, for a basic survey or Bose et al., 2012, for a more recent paper). Beside the k-RNG, Gabriel graphs (Gabriel and Sokal, 1969) and k-NN graphs are prominent examples of proximity graphs.

The 1-RNG, which is simply known as RNG, has been used in a wide range of applications (see Toussaint, 2014, for a review and detailed references). Most interesting for us are its use in classification and clustering as related to our Algorithms 4 and 5, respectively: Instance-based classification based on the RNG neighborhood, that is inferring a point's label by taking a majority vote of the point's neighbors in the RNG, has been empirically shown to be competitive with the k-NN classifier in Sánchez et al. (1997a) and Toussaint and Berzan (2012). Instance-based classification based on the RNG neighborhood has also been used for prototype selection for the 1-NN classifier (Toussaint et al., 1984, Sánchez et al., 1997b). The RNG has been used for spectral clustering in Correa and Lindstrom (2012) with a strategy of assigning locally adapted edge weights. Our experiments in Section 4.5.1 show that such a strategy is dispensable and that using the k-RNG weighted as in (4.8), or also unweighted, yields reasonable results as well.

We have mentioned in Section 4.2.2 and Section 4.3.4 that a true k-RNG (not an estimated one) is always connected. This follows from the fact that the RNG on a data set $\mathcal{D}$ contains the minimal spanning tree on $\mathcal{D}$ as a subgraph. By minimal spanning tree we mean the minimal spanning tree of the complete graph on $\mathcal{D}$ in which an edge is weighted with the distance between two points. A proof of this property for data points in the Euclidean plane, which readily generalizes to data sets in arbitrary semimetric spaces, can be found in Toussaint (1980). The RNG is guaranteed to be sparse for data sets in the 2-dimensional or 3-dimensional Euclidean space, but it can be dense in higher-dimensional spaces or if ι is induced by the 1-norm or the maximum norm

(Jaromczyk and Toussaint, 1992). There is a large literature on the question how to efficiently compute a k-RNG on a data set, mainly for data sets in $\mathbb{R}^2$ or $\mathbb{R}^3$ (see the references in Toussaint, 2014), and how to *approximate* the RNG by a graph that is easier to compute (Andrade and de Figueiredo, 2001). We are not aware of any work that deals with *estimating* the k-RNG as we have done in Section 4.2.2.

4.5 Experiments

We performed several experiments for examining the performance of our proposed Algorithms 1 to 5 and compared them to ordinal embedding approaches. In case of Algorithm 1 and Algorithm 2 we also made a comparison with the methods proposed by Heikinheimo and Ukkonen (2013) explained in Section 4.4.1. We found that our algorithms yield reasonable and useful results, but that for small data sets the embedding approaches tend to be superior in terms of error rates. However, our algorithms are highly superior in terms of computing time (even without making use of their potential of simple and efficient parallelization). The full strength of our algorithms lies in the regime where the ordinal embedding algorithms break down due to computational complexity, but our algorithms still yield useful results.

Recall that an ordinal embedding approach consists of first constructing an ordinal embedding of a data set $\mathcal{D}$ based on the given ordinal data and then solving the problem on the embedding by applying a standard algorithm. For example, in the case of medoid estimation a medoid of an ordinal embedding is computed and the corresponding object is returned as an estimate of a medoid of $\mathcal{D}$. For constructing an ordinal embedding we tried several algorithms: the GNMDS (generalized non-metric multidimensional scaling) algorithm by Agarwal et al. (2007), the SOE (soft ordinal embedding) algorithm by Terada and von Luxburg (2014), and the STE (stochastic triplet embedding) and t-STE (t-distributed stochastic triplet embedding) algorithms by van der Maaten and Weinberger (2012). The GNMDS algorithm and the SOE algorithm can take answers to arbitrary dissimilarity comparisons of the general form (1.1) as input, while the STE and t-STE algorithms are designed only for similarity triplets, that is answers to comparisons (1.2). The ordinal data that we gave to the embedding algorithms were all the similarity triplets obtained via (4.1) from a collection of statements of the kind ($\star$) that we provided as input to one of our algorithms. We used the MATLAB implementations of GNMDS, STE, and t-STE provided by van der Maaten and Weinberger (2012) and the R implementation of SOE provided by Terada and von Luxburg (2014). We set all parameters except the dimension of the space of the embedding to the provided default parameters (for all algorithms the default dimension is two). Note that all algorithms try to iteratively minimize an objective function that measures the amount of violated ordinal relationships, and in doing so their results depend on a random initialization of the ordinal embedding.

We start with presenting experiments using synthetically generated statements of the kind ($\star$) in Section 4.5.1. In Section 4.5.2 we deal with real data consisting of 60 images of cars and statements of the kind ($\star$) that we collected via crowdsourcing.

4.5.1 Synthetically generated statements of the kind ($\star$)

In the following, except the plots in Figures 4.9 and 4.10, in which outliers have to be identified by visual inspection, and one plot in Figure 4.6, which provides a visualization of available statements per data point, all plots of this section show results averaged over 100 runs of an experiment.

We primarily study the performance of the considered methods with respect to the number of provided input statements, but also with respect to the amount of noise in the provided ordinal data. We consider two different noise models: Noise model I (with parameter $0 \leq \mathit{errorprob} \leq 1$) equals the one described in Section 4.2.2, that is a statement of the kind ($\star$) is incorrect, independently of other statements, with some fixed error probability *errorprob*. In an incorrect statement the two data points that are not most central appear to be most central with probability 1/2 each. In Noise model II (with parameter $\mathit{noiseparam} \geq 0$) we distort the dissimilarity values $\iota(A, B)$, which then induces a distortion of statements. Concretely, we add Gaussian noise with mean zero and standard deviation $\mathit{noiseparam} \cdot \mathrm{SD}$, where SD denotes the standard deviation of all true dissimilarity values $\iota(A, B)$, $A \neq B \in \mathcal{D}$, independently to each dissimilarity value $\iota(A, B)$. For choosing input statements we essentially consider two sampling strategies: The first one, referred to as “Uniform sampling”, is to choose input statements uniformly at random without replacement from the set of all statements, that is the set of statements for all triples of data points, which were generated according to the noise model under consideration. When applying this sampling strategy and studying performance as a function of the number of input statements, the rightmost measurement in a plot corresponds to the case that all statements are provided as input. In the experiment presented in Figure 4.8 the provided statements are chosen uniformly at random with replacement from the set of all statements, but there the set of all statements is so large that in fact this does not make any difference. In these plots the rightmost measurement corresponds to a number of input statements of less than one permil of the number of all statements. In order to illustrate our claim that our algorithms require statements to be sampled only approximately uniformly with respect to a fixed data point (Algorithms 1 to 3), or a fixed pair of data points (Algorithm 4 and Algorithm 5), we also consider a second sampling strategy, referred to as “Sampling II”. When sampling according to this strategy, we partition the data set into ten groups. For each group we form a set consisting of all statements, generated according to the noise model under consideration, that comprise at least one data point from the corresponding group. We then sample with replacement by selecting one of the ten sets according to probabilities $p_i = i^2 / \sum_{j=1}^{10} j^2$, $i = 1, \ldots, 10$, and choosing a statement from the selected set uniformly at random.

When comparing Algorithm 1 or Algorithm 2 to the corresponding methods by Heikinheimo and Ukkonen (2013), their methods are given a collection of statements of the kind ($\boxplus$) as input that contains as many statements as the input to our algorithm and is created in a completely analogous way.

Medoid estimation

We measure performance of a method for medoid estimation by the relative error in the objective I (given in (4.5)), which is given by

$$\text{relative error} = \frac{I(\text{estimated medoid}) - I(\text{true medoid})}{I(\text{true medoid})}. \tag{4.17}$$

Figure 4.6 shows in the first two rows the relative error of Algorithm 1, the method by Heikinheimo and Ukkonen (2013), and the ordinal embedding approach, using the various embedding algorithms, as a function of the number of provided input statements and as a function of *errorprob* (Noise model I) for 100 points from a 2-dimensional Gaussian $N_2(0, I_2)$ and ι being the Euclidean metric. Obviously, the embedding approach outperforms Algorithm 1 and the method by Heikinheimo and Ukkonen when dealing only with correct statements, that is $errorprob = 0$, and embedding into the true dimension (1st row, 1st plot). However, it is not superior over Algorithm 1 anymore when $errorprob = 0.3$ and the dimension of the embedding is chosen as five (2nd row, 1st plot). Algorithm 1 consistently outperforms the method by Heikinheimo and Ukkonen. All methods show a similar behavior with respect to *errorprob* (2nd row, 2nd & 3rd plot). Interestingly, the strongest incline in the error does not occur until the transition from $errorprob = 0.6$ to $errorprob = 0.7$. The bottom row of Figure 4.6 also shows the relative error of the various methods as a function of the number of provided input statements, but here input statements were sampled according to the strategy Sampling II. Compared to the strategy of sampling statements uniformly at random without replacement from the set of all statements, Algorithm 1 performs slightly worse, but we consider the difference to be negligible. The last plot of the bottom row shows the difference in the two sampling strategies: while in the uniform case, for all data points there is almost the same number of input statements comprising the data point, when sampling according to Sampling II there are data points for which this number is twice as large as for others (the plot is based on a total of 4500 input statements corresponding to the third measurement in the first and second plot of the bottom row).

The biggest advantage of Algorithm 1 (in fact of *all* our proposed algorithms) compared to an ordinal embedding approach becomes obvious from the plots in the third and fourth row of Figure 4.6, which show the running times of the experiments shown in the plots in the two top rows: For a fixed size $|\mathcal{D}|$ of the data set, like the running times of our proposed algorithms and the method by Heikinheimo and Ukkonen, the running time of the embedding approach with any of the considered embedding algorithms also grows linearly with the number $|\mathcal{S}|$ of input statements (indicated by the orange curves). However, in practice Algorithm 1 and the method by Heikinheimo and Ukkonen are vastly superior in terms of running time compared to the embedding approach, even without making use of their potential of simple and highly efficient parallelization. For example, when all statements are provided as input, $errorprob = 0$, and the embedding dimension is chosen as two, the running time of the embedding approach is between 10 seconds (when using the SOE algorithm) and 141 seconds (when using the t-STE algorithm), while Algorithm 1 or the method by Heikinheimo and Ukkonen only run for 0.01 seconds (3rd row, 1st plot). Note that the running times of Algorithm 1 and the method by Heikinheimo and Ukkonen are independent of *errorprob* and, of course, of the choice

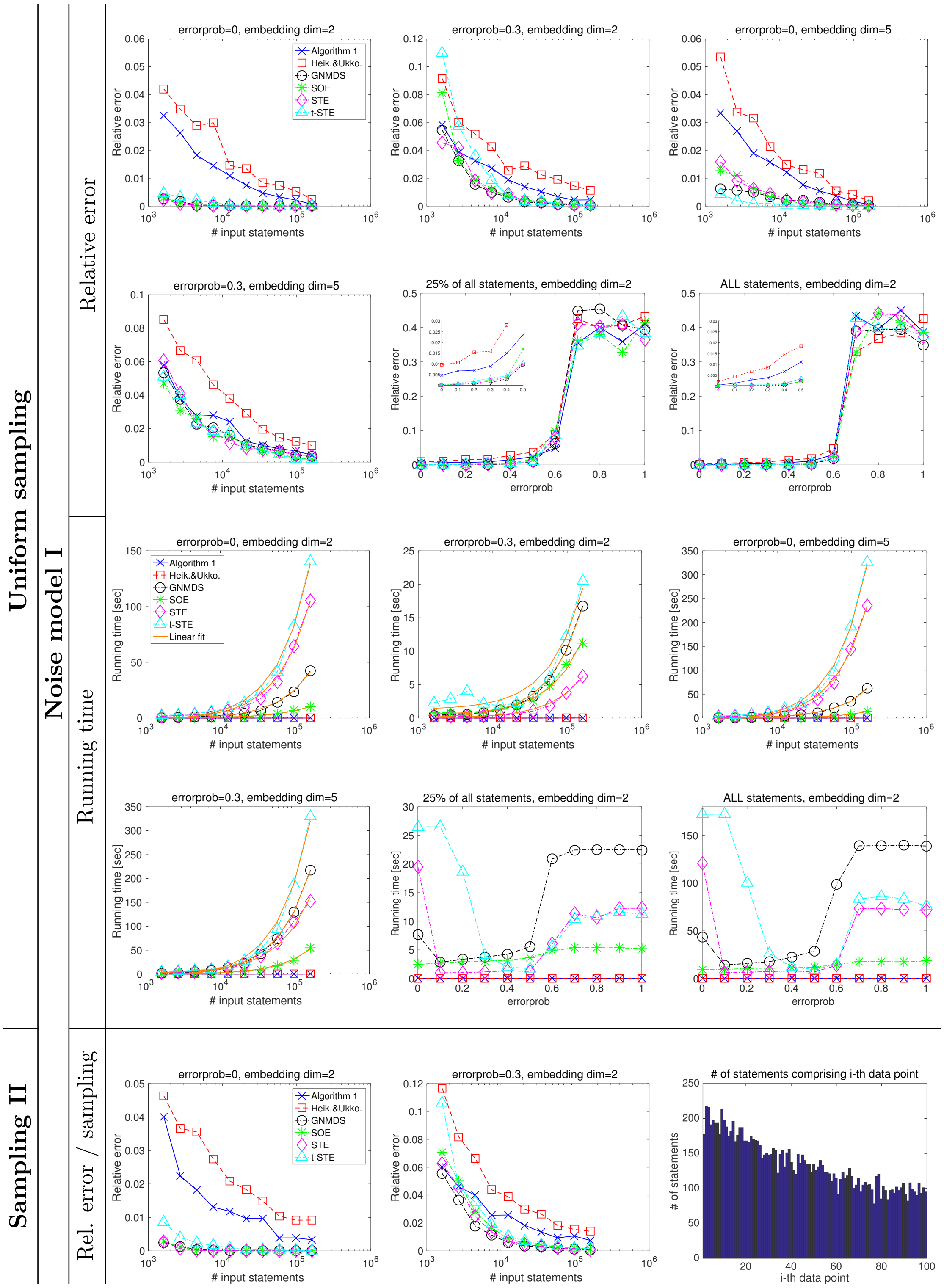


Figure 4.6: Medoid estimation — 100 points from a 2-dim Gaussian $N_2(0, I_2)$ with Euclidean metric. Relative error (4.17) and running time as a function of the number of provided statements of the kind ($\star$) or of the kind ($\boxplus$) and as a function of *errorprob* for Algorithm 1, for the method by Heikinheimo and Ukkonen, and for the embedding approach using the various embedding methods.

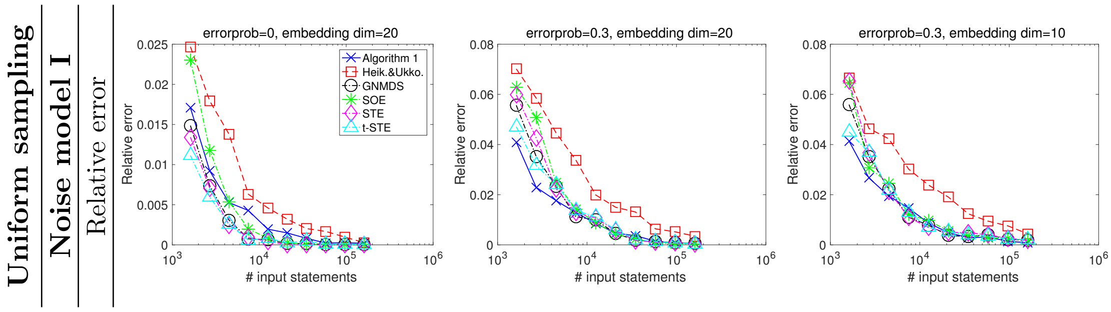


Figure 4.7: Medoid estimation — 100 points from a 20-dim Gaussian $N_{20}(0, I_{20})$ with Euclidean metric. Relative error (4.17) as a function of the number of provided statements of the kind ($\star$) or of the kind ($\boxplus$) for Algorithm 1, for the method by Heikinheimo and Ukkonen, and for the embedding approach using the various embedding methods.

of a dimension of the space of the embedding. The running times of the embedding algorithms tend to increase with the embedding dimension (e.g., differences between the first and the third plot in the third row). The running time of the SOE algorithm also increases with *errorprob* (4th row, 2nd & 3rd plot). For the GNMDS algorithm this holds for *errorprob* ≥ 0.1. The running times of the STE and t-STE algorithms vary non-monotonically with *errorprob*. All experiments shown in Figure 4.6 were performed in MATLAB R2015a on a MacBook Pro with 2.6 GHz Intel Core i7 and 8 GB 1600 MHz DDR3. Within MATLAB we invoked R 3.2.2 for computing the SOE embedding. In order to make a fair comparison we did not use MEX files in the implementation of Algorithm 1 or the method by Heikinheimo and Ukkonen.

Figure 4.7 shows similar experiments as Figure 4.6, but this time we deal with 100 points from a 20-dimensional Gaussian $N_{20}(0, I_{20})$. The distance function ι still equals the Euclidean metric. In this high-dimensional case the embedding approach cannot be considered superior anymore. In fact, when *errorprob* $= 0.3$ and the number of input statements is small, Algorithm 1 performs best (2nd plot). We omit to show plots of the relative error as a function of *errorprob* since they look very similar to the ones in Figure 4.6. Time measurements show that the differences in running times between the embedding approach and Algorithm 1 or the method by Heikinheimo and Ukkonen are even more severe compared to Figure 4.6, as to be expected because of the high embedding dimension (plots omitted).

Finally, we applied Algorithm 1 and the method by Heikinheimo and Ukkonen to a large network with the dissimilarity function ι equaling the shortest-path-distance. In this context, a medoid is usually referred to as a "most central point with respect to the closeness centrality measure" (Freeman, 1978). Our data set consists of 8638 vertices, which form the largest connected component of a collaboration network with 9877 vertices that represent authors of papers submitted to arXiv in the High Energy Physics - Theory category and with two vertices being connected if the authors co-authored at least one paper (Leskovec et al., 2007). Comparing against the embedding methods as in the previous experiments on this large data set would have taken several months (considering various numbers of input statements and averaging over 100 runs), so we only compared against GNMDS (embedding dimension chosen to equal two) for a small number of input statements. The first and the second plot of Figure 4.8 show the

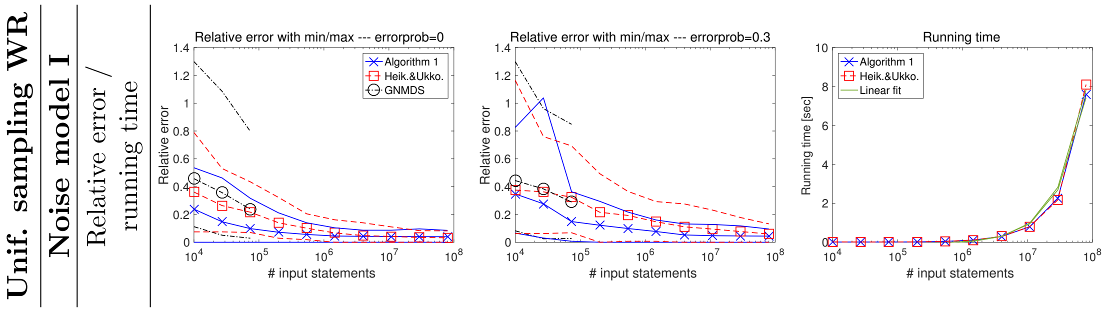


Figure 4.8: Medoid estimation — 8638 vertices in a collaboration network with shortest-path-distance. 1st & 2nd plot: Relative error (4.17) as a function of the number of provided statements of the kind ($\star$) or of the kind ($\boxplus$) for Algorithm 1, for the method by Heikinheimo and Ukkonen, and for the embedding approach using GNMDS (for the first three measurements). Average over 100 runs together with the minimum and the maximum of the 100 runs. 3rd plot: The corresponding running times with fitted linear functions.

relative error of Algorithm 1 and the method by Heikinheimo and Ukkonen as a function of the number of provided statements of the kind ($\star$) or of the kind ($\boxplus$). The number of provided statements varies between 10^4 and $8 \cdot 10^7$. The latter is less than one permil of the number of all statements, which is $\binom{8638}{3} \approx 10^{11}$. This number is so large that the set of all statements does by no means fit into the main memory of a single machine. The plots also show the relative error of the embedding approach using the GNMDS algorithm for 10000, 27144, and 73680 input statements. As in the previous experiments, the shown error is the average over 100 runs of the experiment, but here the data set is fixed and the only sort of randomness comes from the input statements (and the random initialization of the ordinal embedding in case of GNMDS). In addition to the average error the plots show the minimum and maximum error of the 100 runs for illustrating the variance in the methods. In both the cases of $errorprob = 0$ (1st plot) and $errorprob = 0.3$ (2nd plot), when the number of input statements is small, Algorithm 1 outperforms the method by Heikinheimo and Ukkonen. Both methods outperform the embedding approach, which might have difficulties due to the data set being non-Euclidean or might struggle with a too small embedding dimension. For comparison, a strategy of choosing a data point uniformly at random as medoid estimate incurs a relative error of 0.47 in expectation. Even when given only 10000 input statements, when $errorprob = 0$, the error of Algorithm 1 is only about one half of this. The variance seems to be similar for both Algorithm 1 and the method by Heikinheimo and Ukkonen and seems to be significantly larger for the embedding approach. As expected, it decreases as the number of input statements increases. In case of $errorprob = 0$, we also applied GNMDS to the data set providing 10857670 statements as input (corresponding to the eighth measurement in the plots): averaging over 10 runs we obtained an average relative error of 0.28 (which is more than six times larger than the error of Algorithm 1 or the method by Heikinheimo and Ukkonen), where computation took 2.84 hours on average. The third plot of Figure 4.8 shows the running times of Algorithm 1 and the method by Heikinheimo and Ukkonen as a function of the number of input statements. Both methods have the same running time, which is linear in the number of input statements.

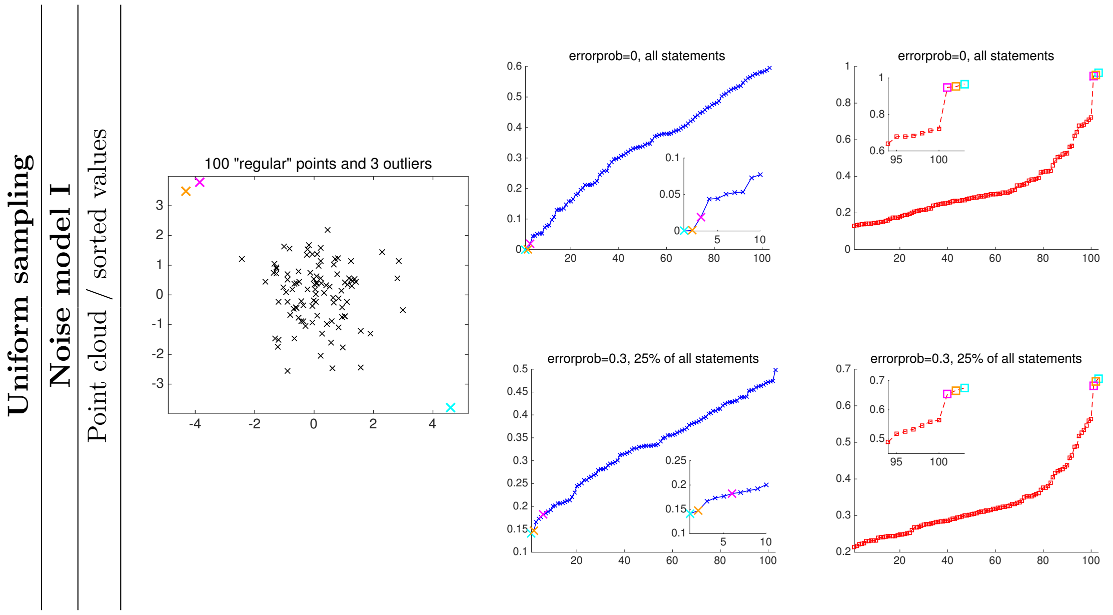


Figure 4.9: Outlier identification — 100 points from a 2-dim Gaussian $N_2(0, I_2)$ and three outliers added by hand with Euclidean metric. Data set and sorted values of $LD(O)$ as needed for Algorithm 2 (left; in blue) and of estimated probabilities as needed for the method by Heikinheimo and Ukkonen (right; in red).

The plot does not show the running times of GNMDS at the first three measurements. These were 110, 860, and 879 seconds in case of $errorprob = 0$ and 92, 848, and 898 seconds in case of $errorprob = 0.3$.

Outlier identification

We started with testing Algorithm 2 and the corresponding method by Heikinheimo and Ukkonen (2013) by applying them to two visualizable data sets containing some obvious outliers. Both of the Figures 4.9 and 4.10 show a scatterplot of the points of a data set $\mathcal{D}$ in the Euclidean plane with the “regular” points in black and the outliers in color. For assessing the performance of the two considered methods we plotted the sorted values of $LD(O)$, $O \in \mathcal{D}$, as needed for Algorithm 2 as well as the sorted values of estimated probabilities of being an outlier within a triple of objects as needed for the method by Heikinheimo and Ukkonen (compare with Section 4.4.1). Both methods were provided with the same number of statements as input, either of the kind ($\star$) or of the kind ($\boxplus$). Both Figure 4.9 and Figure 4.10 provide several such plots, varying with the number of input statements as well as with the error probability $errorprob$ (we generated statements according to Noise model I). In all the plots, values belonging to outliers have the same color as the corresponding outlier in the scatterplot. The methods are successful if these colored values appear at the very end of the sorted values, either at the lower end for Algorithm 2 or at the upper end for the method by Heikinheimo and Ukkonen, and there is a (preferably large) gap between the colored values and the remaining ones since then it is easy to correctly identify the outliers. There are inlay

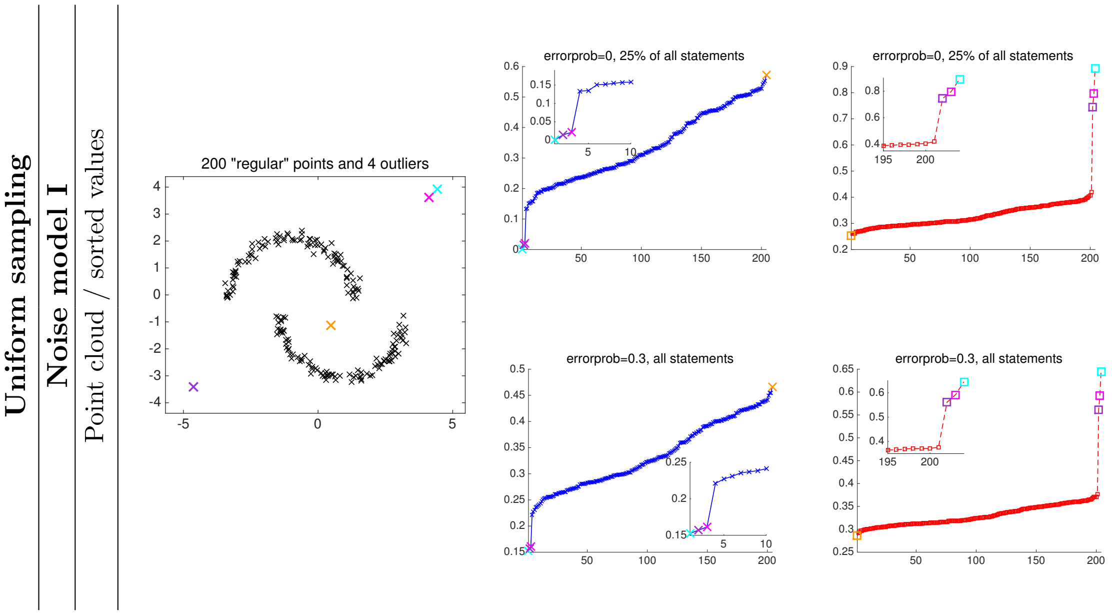


Figure 4.10: Outlier identification — 200 **points from a Two-moons data set and four outliers added by hand with Euclidean metric.** Data set and sorted values of $LD(O)$ as needed for Algorithm 2 (left; in blue) and of estimated probabilities as needed for the method by Heikinheimo and Ukkonen (right; in red).

plots showing the bottom or top ten values for more precise inspection. Note that there is no averaging involved in creating these plots and they may change with every run of the experiment since they depend on the random data set, the random choice of statements that are provided as input, and the random occurrence of incorrect statements.

In Figure 4.9 the data set consists of 100 points that were drawn from a 2-dimensional Gaussian $N_2(0, I_2)$ and three outliers added by hand. The dissimilarity function ι equals the Euclidean metric. We can see that for both methods the values corresponding to the outliers appear at the right place when given all correct statements as input (top row). However, when given only 25 percent of all statements and $errorprob = 0.3$, for Algorithm 2 the estimated lens depth value of the pink outlier ranks only sixth smallest, and thus this outlier might not be identified (bottom left). Furthermore, even in the previous situation it might not be possible to correctly infer the number of outliers based on the plot corresponding to Algorithm 2 due to the lack of a clear gap, whereas in both situations this can easily be done for the method by Heikinheimo and Ukkonen. We made similar observations for smaller numbers of provided input statements and other values of $errorprob$ too (plots omitted).

In Figure 4.10 the data set consists of 200 points from a Two-moons data set and four outliers added by hand. Again, ι equals the Euclidean metric. Both methods correctly identify the three outliers located quite far apart from the bulk of the data points, and the gap between their values and values belonging to the "regular" data points is large enough to be easily spotted. However, both methods fail to identify the outlier located in-between the two moons (yellow point). The estimated lens depth values or probabilities indicate that this outlier might be the unique medoid—which is indeed the case. For

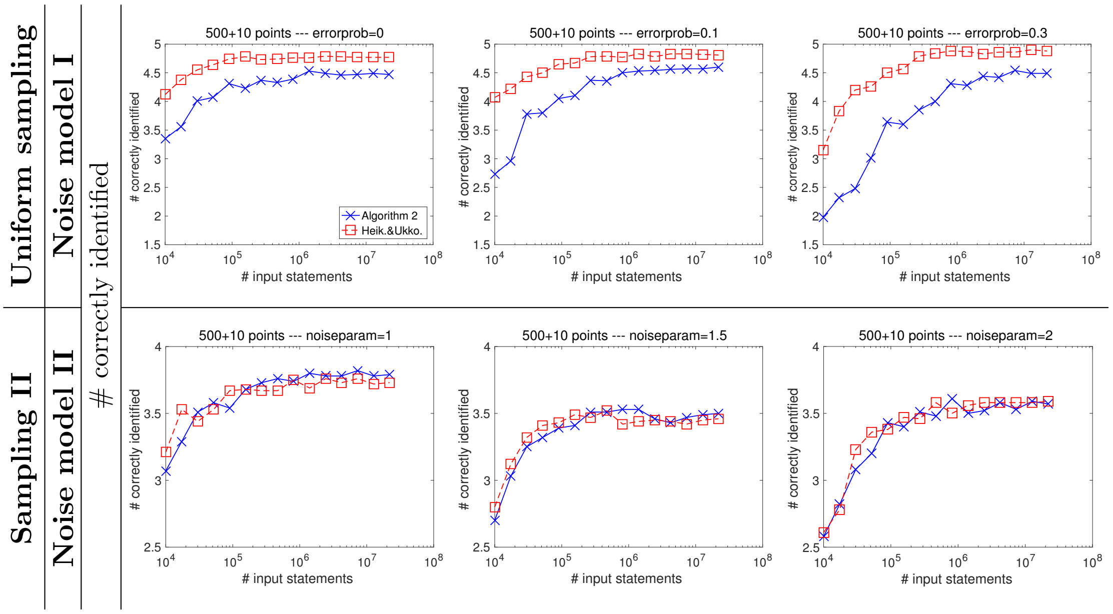


Figure 4.11: Outlier identification — 500 points from the subset of USPS digits 6 and ten outlier digits with Euclidean metric. Number of correctly ranked outliers as a function of the number of provided statements of the kind $(\star)$ or of the kind $(\boxplus)$ for Algorithm 2 and for the method by Heikinheimo and Ukkonen.

Algorithm 2 this has to be expected and stresses the inherent property of the lens depth function, and statistical depth functions in general, of globally measuring centrality. In doing so, it ignores multimodal aspects of the data (compare with Section 4.4.2 and Section 4.6) and cannot be used for identifying outliers that are globally seen at the heart of a data set. At least for the data set of Figure 4.10 this also holds for the function F defined in (4.16), which the method by Heikinheimo and Ukkonen is based on. However, for the function F this behavior is not systematic as the example of a symmetric bimodal distribution in one dimension as mentioned in Section 4.4.2 shows.

In the last experiment of this section we study Algorithm 2 and the method by Heikinheimo and Ukkonen by using them for outlier identification in a data set consisting of USPS digits. The data set consists of 500 digits chosen uniformly at random from digits 6 and ten outlier digits chosen uniformly at random from the remaining digits. The dissimilarity function ι equals the Euclidean metric. We assess the performance of Algorithm 2 and the method by Heikinheimo and Ukkonen by counting how many of the ten outliers are among the ten digits ranked lowest or highest according to the values of $LD(O)$ and estimated probabilities, respectively. Figure 4.11 shows these numbers as a function of the number of provided input statements in case of uniform sampling and statements generated according to Noise model I (1st row) and in case of Sampling II and statements generated according to Noise model II (2nd row), for $errorprob = 0$ / $noiseparam = 1$ (1st plot), $errorprob = 0.1$ / $noiseparam = 1.5$ (2nd plot), and $errorprob = 0.3$ / $noiseparam = 2$ (3rd plot). We can see that the method by Heikinheimo and Ukkonen performs slightly better in the setting of the first row and that the performance of both methods is essentially the same in the setting of the second row. Most often, the methods can identify three to five outliers, which we consider to be not bad, but not good either. Choosing another digit than 6 for defining the bulk of "regular" points leads to similar results (plots omitted).

To sum up the insights from the experiments shown in Figures 4.9 to 4.11, we may conclude that both Algorithm 2 and the method by Heikinheimo and Ukkonen are capable of identifying outliers located lonely and far apart from the bulk of a data set, but should be used with some care in general. The method by Heikinheimo and Ukkonen seems to be superior—which is not very surprising since statements of the kind ($\boxplus$) readily inform about outliers within triples of data points. It produces larger and thus easier to spot gaps than Algorithm 2, but is less understood theoretically.

Classification

We compared Algorithms 3 and 4 to an ordinal embedding approach that consists of embedding a data set $\mathcal{D}$ comprising a set $\mathcal{L}$ of labeled data points and a set $\mathcal{U}$ of unlabeled data points into $\mathbb{R}^d$ using the given ordinal distance information and applying a classification algorithm to the embedding. Note that this approach is semi-supervised since it makes use of answers to dissimilarity comparisons involving data points of $\mathcal{U}$ for constructing the embedding of $\mathcal{D}$. Algorithm 3, in contrast, only uses ordinal distance information involving data points of $\mathcal{L}$ for approximately evaluating the feature map (4.6) on $\mathcal{L}$ and hence is a supervised technique as long as the classifier on top is. Algorithm 4 is a supervised instance-based learning method. Algorithm 3 as well as the embedding approach require an ordinary classifier on top, that is a classifier appropriate for real-valued feature vectors. For simplicity, in the experiments presented here we either used the k-NN classifier or the SVM (support vector machine) algorithm with the standard linear kernel (e.g., Cristianini and Shawe-Taylor, 2000). Both these classification algorithms require to set parameters, which we did by means of 10-fold cross-validation: the parameter k for the k-NN classifier was chosen from the range $1, 3, 5, 7, 11, 15, 23$ and the regularization parameter for the SVM algorithm was chosen from $0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000$. The ordinal embedding algorithms produce embeddings on an arbitrary scale. Before applying the classification algorithms, we rescaled the produced ordinal embeddings to have diameter 2. The feature embedding constructed by Algorithm 3 always resides in $[0, 1]^K$ for a K-class classification problem and no rescaling was done here. Algorithm 4 requires to set the parameter k describing which k-RNG it is based on, but this is more subtle: As we have seen in Section 4.2.2, when input statements are incorrect with some error probability $errorprob > 0$ (Noise model I), then our estimation strategy does not estimate the k-RNG anymore, but rather a k'-RNG with $k' = k'(k, errorprob, |\mathcal{D}|)$ depending on the size of the data set as given in (4.14). We thus have to choose the range of possible values for the parameter k in Algorithm 4 depending on $|\mathcal{D}|$. Furthermore, we cannot use 10-fold cross-validation for choosing the best value within this range since, roughly speaking, this would lead to choosing the best parameter for a data set of size of only 90 percent of $|\mathcal{D}|$. Instead, we used a non-exhaustive variant of leave-one-out cross-validation: we randomly selected a single training point as validation set and repeated this procedure for 20 times, and finally chose the parameter that showed the best performance on average.

We measure performance of Algorithms 3 and 4 and the embedding approach by

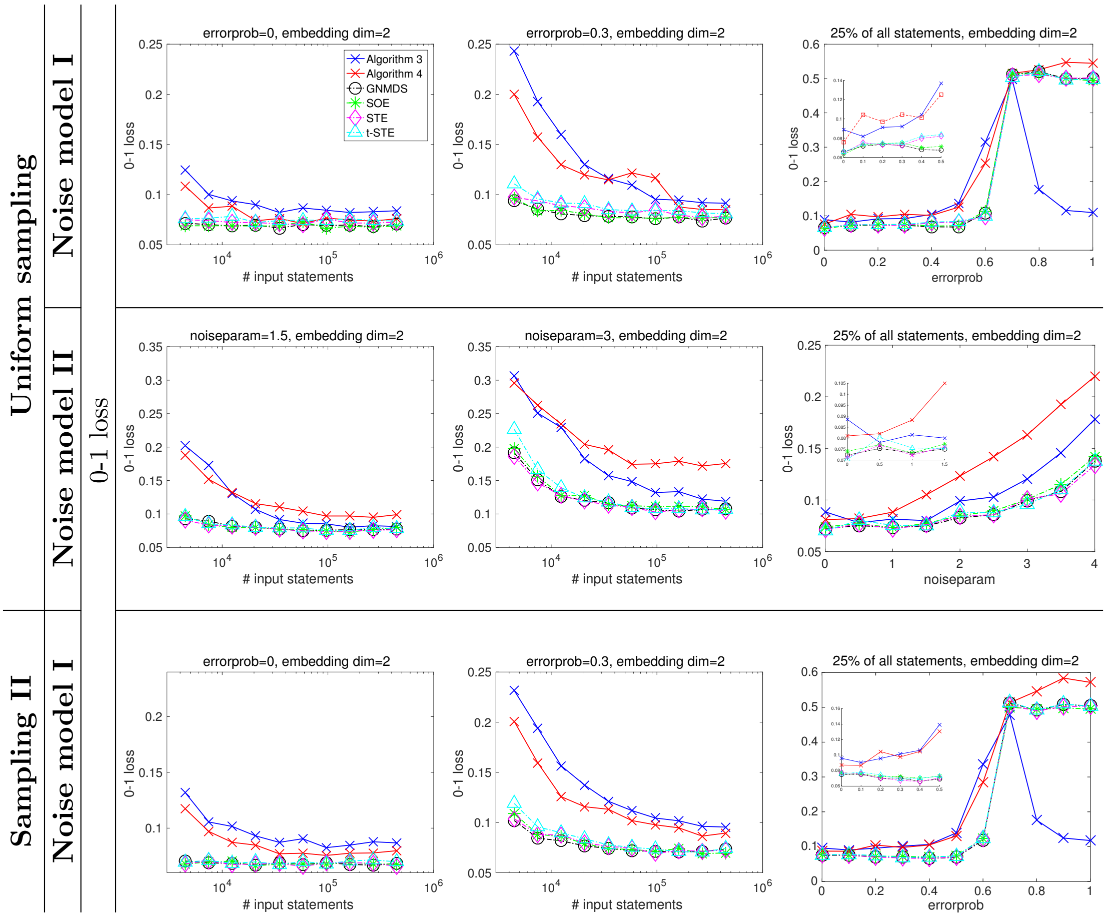


Figure 4.12: Classification — 100 labeled and 40 unlabeled points from a mixture of two equally probable 2-dim Gaussians $N_2(0, I_2)$ and $N_2((3,0)^T, I_2)$ with Euclidean metric. SVM algorithm with linear kernel on top of Algorithm 3 as well as on the embedding approach. 0-1 loss (4.18) as a function of the number of provided statements of the kind ($\star$) and as a function of *errorprob* / *noiseparam* for Algorithm 3, for Algorithm 4, and for the embedding approach using the various embedding methods.

considering their incurred 0-1 loss given by

$$\text{0-1 loss} = \frac{1}{|\mathcal{U}|} \cdot \sum_{O \in \mathcal{U}} \mathbb{1}\{\text{predicted label}(O) \neq \text{true label}(O)\}. \qquad (4.18)$$

Figure 4.12 shows the results for a data set consisting of 100 labeled and 40 unlabeled points from a mixture of two equally probable 2-dimensional Gaussians $N_2(0, I_2)$ and $N_2((3,0)^T, I_2)$ and ι being the Euclidean metric. True class labels of the points correspond to which Gaussian they come from. On top of Algorithm 3 as well as on the embedding methods we used the SVM algorithm with the linear kernel. The parameter k for Algorithm 4 was chosen from the range $1, 2, 3, 5, 7, 15, 25, 45, 70$. The dimension of the space of the ordinal embedding was chosen to equal the true dimension two, but we observed similar results when we chose it as five instead (plots omitted). The ordinal embedding approach outperforms both Algorithm 3 and Algorithm 4, but their results

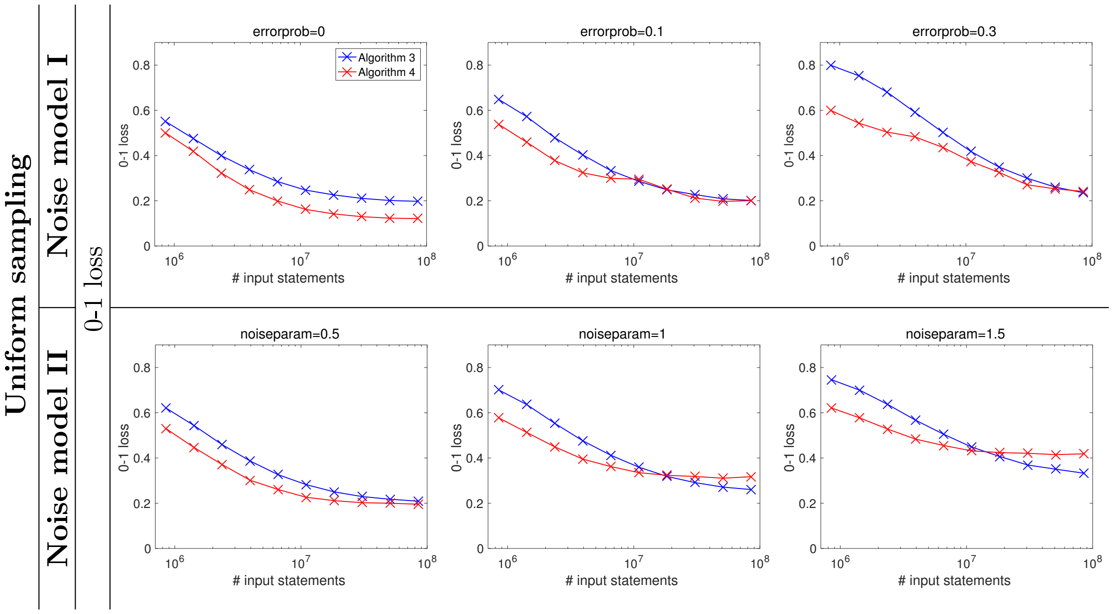


Figure 4.13: Classification — 300 labeled and 500 unlabeled USPS digits with Euclidean metric. k-NN classifier on top of Algorithm 3. 0-1 loss (4.18) as a function of the number of provided statements of the kind ($\star$) for Algorithms 3 and 4.

appear to be acceptable too. Interestingly, other than for Algorithm 4 and the embedding approach, the 0-1 loss incurred by Algorithm 3 studied as a function of $errorprob$ (1st & 3rd row, 3rd plot) increases only up to $errorprob = 0.7$ and then drops again, finally yielding almost the same result for $errorprob = 1$ as for $errorprob = 0$. In hindsight, this is not surprising: If $errorprob = 1$, and thus every statement is incorrect, and the two possibilities of an incorrect statement are equally likely, as it is the case under Noise model I, then Algorithm 3 approximately evaluates the feature map

$$x \mapsto \left(\frac{1}{2} - \frac{1}{2}LD(x; Class_1), \frac{1}{2} - \frac{1}{2}LD(x; Class_2), \ldots, \frac{1}{2} - \frac{1}{2}LD(x; Class_K)\right) \in \mathbb{R}^K.$$

This feature map coincides with the original one given in (4.6) up to a similarity transformation and hence gives rise to the same classification results.

In Figure 4.13 we study the performance of Algorithms 3 and 4 when used for classifying USPS digits. We deal with 800 digits chosen uniformly at random from the set of all USPS digits and randomly split into 300 labeled and 500 unlabeled data points. The dissimilarity function ι equals the Euclidean metric. We chose input statements uniformly at random without replacement from the set of all statements, which we generated according to Noise model I (1st row) or Noise model II (2nd row). On top of Algorithm 3 we used the k-NN classifier. The parameter k for Algorithm 4 was chosen from $1, 2, 3, 5, 7, 15, 25, 45, 70, 100, 150, 230, 350$. For small values of $errorprob$ or $noiseparam$ we consider the results of our proposed algorithms to be satisfactory and useful. Note that in this 10-class classification problem a strategy of random guessing would yield a 0-1 loss of about 0.9. Not surprisingly, we obtained slightly better results when the ratio between labeled and unlabeled data points was chosen as 400/400 instead

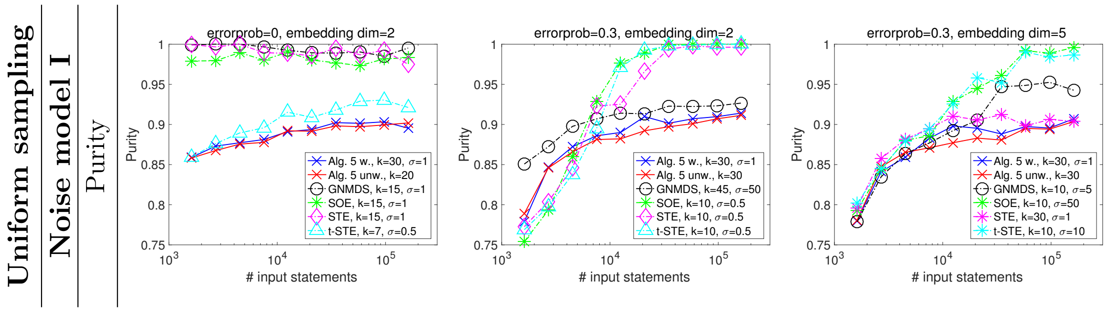


Figure 4.14: Clustering — 100 points from a uniform distribution on two equally sized moons in $\mathbb{R}^2$ with Euclidean metric. Purity (4.19) as a function of the number of provided statements of the kind ($\star$) for Algorithm 5 in its weighted and unweighted version and for the embedding approach using the various embedding methods.

of 300/500 and slightly worse results when it was chosen as 200/600 (plots omitted).

Clustering

In this section we study the performance of Algorithm 5, both in its weighted and in its unweighted version, and compare it to an embedding approach in which we apply spectral clustering to a symmetric k-NN graph on an ordinal embedding of a data set $\mathcal{D}$. The edges of this k-NN graph are weighted by Gaussian weights $\exp(-\|u_i - u_j\|^2/\sigma^2)$, where u_i and u_j are connected points of the embedding, which is rescaled to have diameter 2, and $\sigma > 0$ is a scaling parameter. For assessing the quality of a clustering we measure its purity. Purity (e.g., Manning et al., 2008, Chapter 16) is a widely used external criterion for assessing clustering quality, that is a measure of accordance between an inferred clustering and a known ground truth partitioning of the data set $\mathcal{D}$. If $\mathcal{D}$ consists of L different classes $C_1, \ldots, C_L$ that we would like to recover and the clustering $\mathcal{C}$ comprises K different clusters $U_1, \ldots, U_K$, then the purity of $\mathcal{C}$ is given by

$$\text{purity}(\mathcal{C}) = \text{purity}(\mathcal{C}; \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{k=1}^{K} \max_{l=1,\ldots,L} |U_k \cap C_l|. \tag{4.19}$$

We always have $K/|\mathcal{D}| \leq \text{purity}(\mathcal{C}) \leq 1$, and a high value indicates a good clustering. In the experiments presented in this section, we always provided Algorithm 5 and the ordinal embedding approach with the correct number L of clusters as input. Both in Algorithm 5 and in the embedding approach we use the normalized version of spectral clustering as stated in von Luxburg (2007) and invented by Shi and Malik (2000).

Figure 4.14 shows the purity of the clusterings produced by Algorithm 5 and the embedding approach when applied to a data set consisting of 100 points from a uniform distribution on two equally sized moons in $\mathbb{R}^2$. The two moons correspond to a ground truth partitioning into two classes. The dissimilarity function ι equals the Euclidean metric. The curves shown here are the results obtained by a particular choice of input parameters k and σ: within a reasonably large range of parameter configurations this choice of parameters yielded the best performance on average with respect to the number

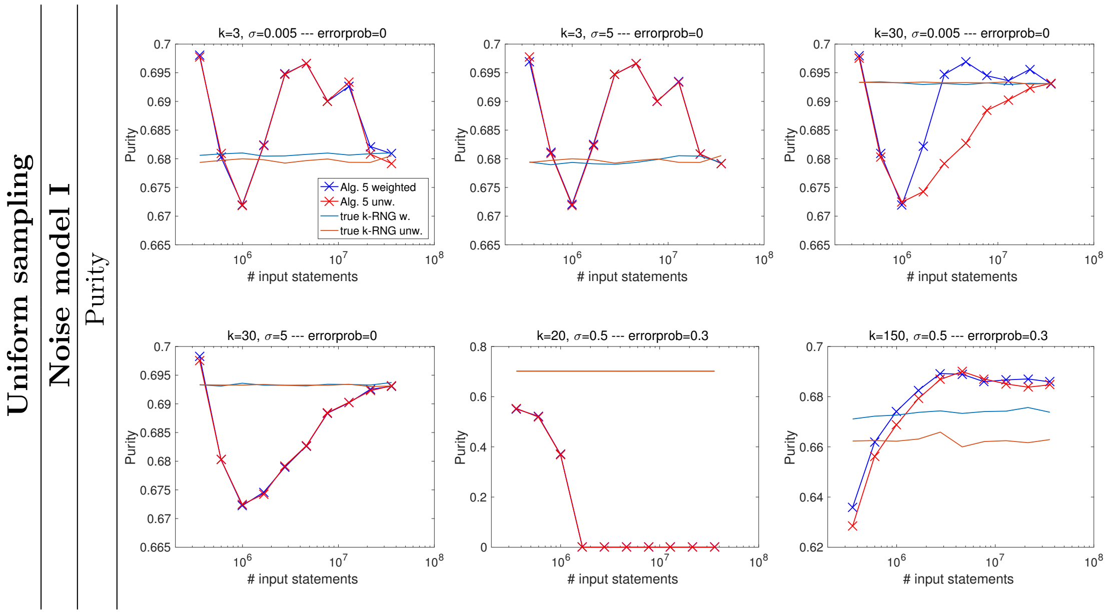


Figure 4.15: Clustering — 600 USPS digits with Euclidean metric. Purity (4.19) as a function of the number of provided statements of the kind ($\star$) for Algorithm 5 in its weighted and unweighted version. The light blue curve and the bronze curve show the purity of the clusterings obtained by applying spectral clustering to the true weighted and unweighted k-RNG on the data set.

of input statements. We study the sensitivity of Algorithm 5 with respect to the parameters in another experiment (shown in Figure 4.15). The embedding approach clearly outperforms Algorithm 5 if $errorprob = 0$ (1st plot), where three of the considered embedding algorithms achieve significantly higher purity values over the whole range of the number of input statements. However, given the numerous advantages common to all our proposed algorithms compared to an ordinal embedding approach, we consider the performance of Algorithm 5 to be acceptable. For comparison, a random clustering in which data points are randomly assigned to one of two clusters independently of each other with probability one half has an average purity of 0.54. If $errorprob = 0.3$, the embedding approach is superior to Algorithm 5 only if the number of input statements is large. Interestingly, there is almost no difference in the performance of the weighted and the unweighted version of Algorithm 5.

The experiment shown in Figure 4.15 deals with a data set $\mathcal{D}$ consisting of 600 digits chosen uniformly at random from the set of all USPS digits and ι being the Euclidean metric. We assume a ground truth partitioning of $\mathcal{D}$ into ten classes according to the digits' values. For various parameter configurations the plots show the purity of the clusterings produced by the weighted (in blue) and unweighted (in red) version of Algorithm 5 as a function of the number of input statements. The plots also show the purity of the clusterings obtained when applying spectral clustering to the true weighted (in light blue) and unweighted (in bronze) k-RNG on $\mathcal{D}$. Note that these two curves only vary with the number of input statements because of random effects in the K-means step of spectral clustering. Although it might look odd at a first glance that the

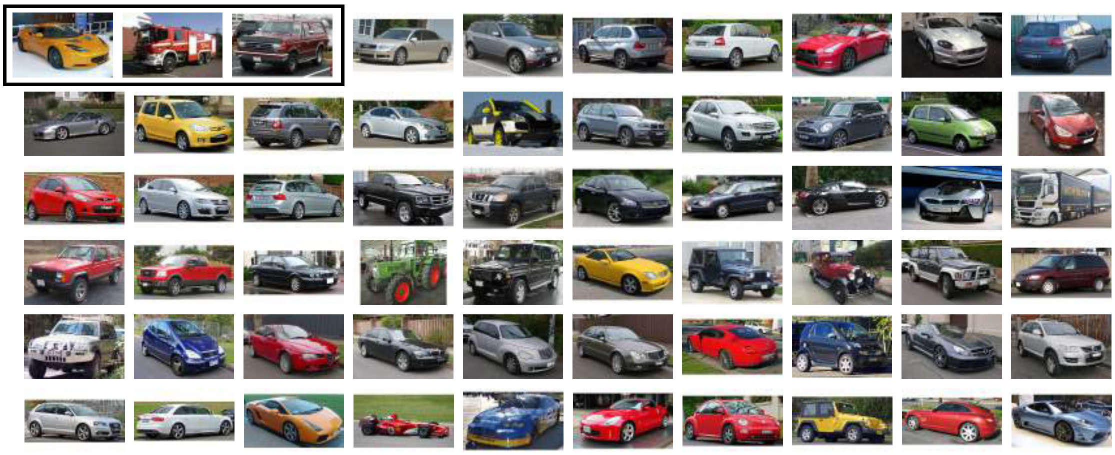


Figure 4.16: Car data set. We collected ordinal distance information of the kind ($\star$) for this data set via an online survey. The framed triple in the first row was used as a test case: T_All and T_All_reduced comprise only statements provided by participants that chose the off-road vehicle as the most central car in this triple.[2]

purity achieved by Algorithm 5 is not monotonic with respect to the number of input statements, we can see that the purity is always between 0.67 and 0.7 for a wide range of values of k and σ when *errorprob* $= 0$ (1st row; 2nd row, 1st plot). For comparison, a random clustering in which data points are randomly assigned to one of ten clusters independently of each other with probability one-tenth has an average purity of 0.19. A clustering obtained by applying spectral clustering to a symmetric k-NN graph with Gaussian edge weights $\exp(-\iota(x_i, x_j)^2/\sigma^2)$ on $\mathcal{D}$ (the true data set—not an ordinal embedding) has an average purity of not higher than 0.75, even for a good choice of k and σ. When *errorprob* $= 0.3$, Algorithm 5 completely fails for small values of k (2nd row, 2nd plot) as has to be expected because of our findings in Section 4.2.2. For k sufficiently large it finally yields the same purity values as when *errorprob* $= 0$ (2nd row, 3rd plot). In fact, this is true already for $k = 100$ and a wide range of values of σ (plots omitted). Again, both in the case of *errorprob* $= 0$ and in the case of *errorprob* $= 0.3$, there is almost no difference in the performance of the weighted and the unweighted version of Algorithm 5.

4.5.2 Crowdsourced statements of the kind ($\star$)

We set up an online survey for collecting ordinal distance information of the kind ($\star$) for 60 images of cars, shown in Figure 4.16. All images were found on Wikimedia Commons (`https://commons.wikimedia.org`) and have been explicitly released into the public domain by their authors. We refer to the set of these images as the car data set. We instructed participants of the survey to determine the most central object within a triple of three shown images according to how they perceive dissimilarity between *cars*.

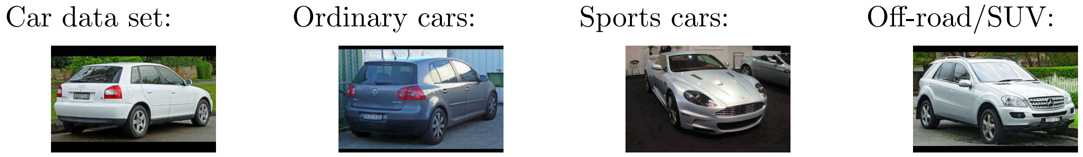
[2] All pictures were found on Wikimedia Commons and have been explicitly released into the public domain by their authors.

Table 4.1: Characteristic values of All, All_reduced, T_All, and T_All_reduced.

	All	All_reduced	T_All	T_All_reduced
Number of statements	7097	6338	6757	6056
Number of statements in percent of number of triples [$\binom{60}{3} = 34220$]	20.74*	18.52	19.75*	17.70
Average number of statements in which a car appears	354.85	316.90	337.85	302.80
Minimum number of statements in which a car appears	307	286	292	269
Maximum number of statements in which a car appears	503	347	478	333
Median response time per shown triple (in seconds)	4.02		4.15	

* Note that All and T_All contain repeatedly present and contradicting statements.

We explicitly stated that they should not judge differences between the *pictures*, like perspective, lighting conditions, or background. Every participant was shown triples of cars in random order (more precisely, triples shown to a participant were drawn uniformly at random without replacement from the set of all possible triples). Also the order of cars within a triple, that is whether a car's image appeared to the left, in the middle, or to the right, was random. One complete round of the survey consisted of 50 shown triples, but we encouraged participants to contribute more than one round, possibly at a later time. There was no possibility of skipping triples, that is even if a participant had no idea which car might be the most central one in a triple, he/she had to make a choice—or quit the current round of the survey. Within the first ten triples every participant was shown a test case triple (shown within a frame in Figure 4.16), consisting of an off-road vehicle, a sports car, and a fire truck. We believe the off-road vehicle to be the obvious most central car in this triple and used this test case for checking whether a participant might have got the task of choosing a most central object correctly. The survey was online for about two months and the link to the survey was distributed among colleagues and friends. We took no account of rounds of the survey that were quitted before 30 triples (of the fifty per round) were shown. In doing so, we ended up with 146 rounds (some of them not fully completed) and a total of 7097 statements. It is hard to guess how many different people contributed to these 146 rounds, but assuming an average of three to four rounds per person, which seems to be reasonable according to personal feedback, their number should be around 40. In only 7 out of the 146 rounds the off-road vehicle was not chosen as most central car in the test case triple. We refer to the collection of all 7097 statements as the collection All and to its subcollection comprising 6757 statements gathered in the 139 rounds in which the off-road vehicle was chosen as most central car in the test case triple as the collection T_All. From All and T_All we derived two more collections of statements of the kind ($\star$) for the



Car data set: Ordinary cars: Sports cars: Off-road/SUV:

Figure 4.17: The estimated medoids for the car data set and the subclasses of ordinary cars, sports cars, and off-road/sport utility vehicles when working with the statements in All or T_All.

car data set as follows: All_reduced is obtained from All by replacing all statements dealing with the same triple of cars by just one statement about this triple, with the most central car being that car that is most often the most central car in the statements to be replaced. T_All_reduced is derived from T_All analogously. The characteristic values of the four collections All, All_reduced, T_All, and T_All_reduced are summarized in Table 4.1.

We applied Algorithms 1 to 5 to the car data set and the statements in All, All_reduced, T_All, or T_All_reduced. In doing so, we assumed a partitioning of the car data set into four subclasses: ordinary cars, sports cars, off-road/sport utility vehicles, and outliers. We considered the fire truck, the motortruck, the tractor, and the antique car as outliers. Looking at Figure 4.16, there should be no doubts about the other classes.

Medoid estimation

We applied Algorithm 1 to the car data set as well as to the three classes of ordinary cars, sports cars, and off-road/sport utility vehicles in order to estimate a medoid within these subclasses and the statements in All, All_reduced, T_All, or T_All_reduced. The estimated medoids obtained when working with All or T_All coincide and are shown in Figure 4.17. The estimated medoids obtained when working with All_reduced or T_All_reduced differ from these only for the whole car data set and the subclass of off-road/sport utility vehicles. Note that for estimating a medoid of a subset of a data set we consider only statements comprising three objects of the subset. For example, when estimating a medoid of the subclass of sports cars based on the statements in All, we effectively work with 89 out of the 7097 statements in All.

It is interesting to study an ordinal embedding of the car data set. Figure 4.18 shows such an embedding into the two-dimensional plane, which we computed with the SOE algorithm based on the statements in T_All. We cannot only see a grouping of the cars according to the subclasses (compare with the section on clustering below) and the outer positioning of the outliers (compare with the following section on outlier identification), but also that our medoid estimates are located quite at the center of the corresponding subclasses (with the exception of the subclass of off-road/sport utility vehicles). This confirms the plausibility of our estimates. Note that we observe slightly different embeddings depending on the random initialization in the SOE algorithm. Also, it is not useful to compare the medoid estimates of Algorithm 1 with estimates based on an ordinal embedding since the latter change with every run of the embedding algorithm.

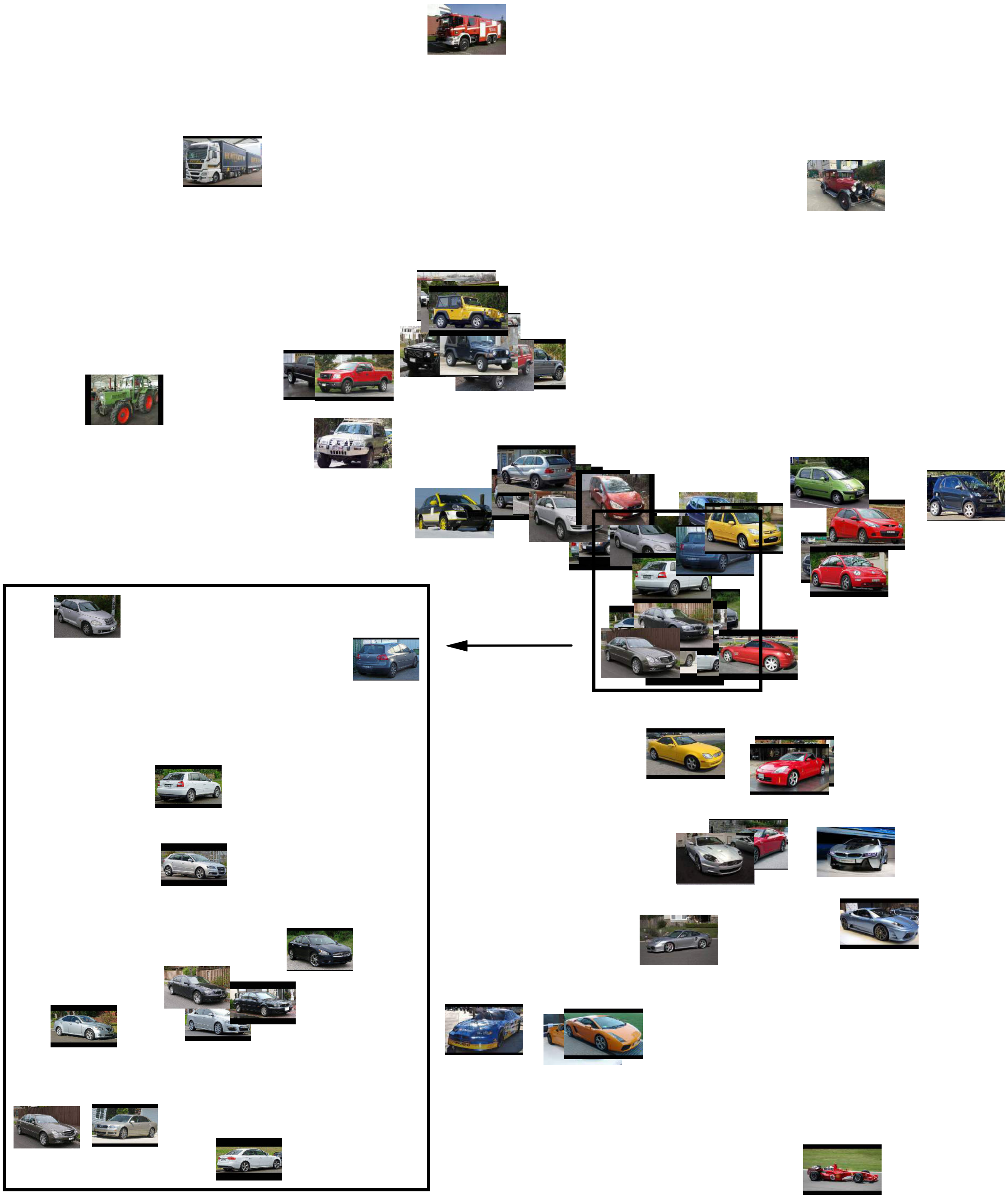


Figure 4.18: An ordinal embedding of the car data set based on the statements in T_All.

Outlier identification

We applied Algorithm 2 to the car data set and the statements in All, All_reduced, T_All, and T_All_reduced, respectively. For all of the four collections of statements we obtained very similar results. Figure 4.19 shows a plot of the sorted values $LD(O)$

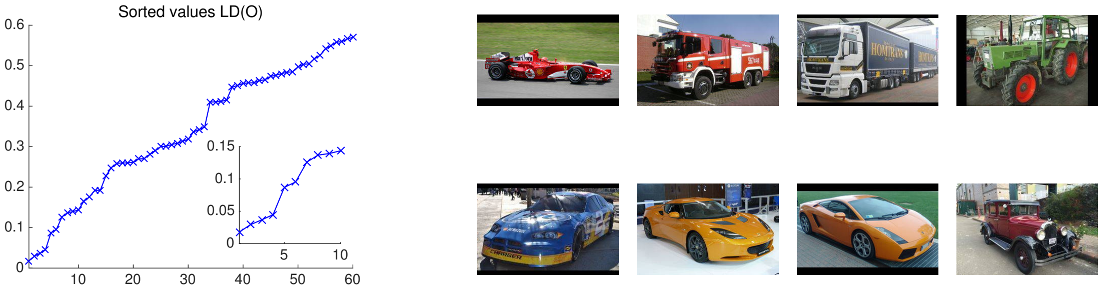


Figure 4.19: The sorted values $LD(O)$ (O in car data set) as well as the eight cars with smallest values (increasingly ordered) when working with the statements in T_ALL.

(for O being an element of the car data set) as well as the eight cars with smallest values when working with T_ALL. Looking at the plot it might be reasonable to assume that there are at least four outliers. Indeed, the Formula One car, the fire truck, the motortruck, and the tractor, which appear rather odd in the car data set, are ranked lowest. Also the other cars shown in Figure 4.19 are quite out of character for the car data set. In the ordinal embedding shown in Figure 4.18 all these cars are located far outside. These findings support our claim that Algorithm 2 might be useful for outlier identification when given only ordinal distance information of the kind ($\star$).

Classification

For setting up a classification problem on the car data set we removed the four outliers (the fire truck, the motortruck, the tractor, and the antique car) and assigned a label to the remaining cars according to which of the three classes of ordinary cars, sports cars, or off-road/sport utility vehicles they belong to. By removing from the collections ALL, ALL_REDUCED, T_ALL, and T_ALL_REDUCED all statements that comprise one or more outliers, we obtained collections of statements of the kind ($\star$) for these 56 labeled cars. A bit sloppy, from now on till the end of Section 4.5.2, by ALL, ALL_REDUCED, T_ALL, and T_ALL_REDUCED we mean these newly created, reduced collections. Their sizes are given in Table 4.2.

We randomly selected 16 cars that we used as test points, that is we ignored their labels and predicted them by applying Algorithms 3 and 4 and an embedding approach based on the label information of the remaining 40 labeled cars and the ordinal distance information in ALL, ALL_REDUCED, T_ALL, or T_ALL_REDUCED. In Table 4.3 we report the average 0-1 loss (see equation (4.18) for its definition) and its standard deviation, where the average is over hundred random selections of test points, for the considered methods and various classification algorithms on top of Algorithm 3 or the embedding approach. We chose the dimension of the space of the embedding as two. As classifiers on top we used both the k-NN classifier and the SVM algorithm, the latter with the linear as well as with the Gaussian kernel. Since we are dealing with a 3-class classification problem, we combined the SVM algorithm with a one-vs-all strategy. We chose the parameter k for the k-NN classifier and the regularization parameter for the SVM algorithm by means of 10-fold cross-validation from $1, 3, 5, 7, 11, 15$

Table 4.2: Number of statements after removing the fire truck, the motortruck, the tractor, and the antique car from the car data set.

	All	All_reduced	T_All	T_All_reduced
Number of statements	5624	5121	5349	4886
Number of statements in percent of number of triples $[\binom{56}{3} = 27720]$	20.29*	18.47	19.30*	17.63

* All and T_All contain repeatedly present and contradicting statements.

Table 4.3: Average 0-1 loss (± standard deviation) when predicting labels for a randomly chosen subset of 16 cars (average over 100 choices).

	All	All_reduced	T_All	T_All_reduced
Alg. 3 with k-NN	0.19 (± 0.11)	0.23 (± 0.12)	0.16 (± 0.10)	0.17 (±0.10)
Alg. 3 with SVM linear	0.17 (± 0.09)	0.16 (± 0.09)	0.13 (± 0.07)	0.16 (± 0.08)
Alg. 3 with SVM Gauss	0.17 (± 0.10)	0.18 (± 0.10)	0.14 (± 0.10)	0.16 (± 0.09)
Algorithm 4	0.15(± 0.09)	0.18 (± 0.10)	0.13 (± 0.09)	0.13 (±0.09)
GNMDS with k-NN	0.05 (± 0.05)	0.05 (± 0.05)	0.04 (± 0.04)	0.04 (±0.04)
GNMDS with SVM linear	0.07 (± 0.07)	0.06 (± 0.06)	0.07 (± 0.07)	0.04 (± 0.05)
GNMDS with SVM Gauss	0.05 (± 0.06)	0.04 (± 0.05)	0.04 (± 0.05)	0.04 (± 0.05)
SOE with k-NN	0.06 (± 0.05)	0.07 (± 0.05)	0.06 (± 0.06)	0.07 (±0.06)
SOE with SVM linear	0.10 (± 0.08)	0.12 (± 0.09)	0.11 (± 0.08)	0.10 (± 0.09)
SOE with SVM Gauss	0.05 (± 0.07)	0.07 (± 0.08)	0.05 (± 0.05)	0.07 (± 0.06)
STE with k-NN	0.05 (± 0.04)	0.03 (± 0.04)	0.05 (± 0.04)	0.04 (± 0.04)
STE with SVM linear	0.07 (± 0.08)	0.06 (± 0.06)	0.08 (± 0.07)	0.06 (± 0.06)
STE with SVM Gauss	0.05 (± 0.05)	0.03 (± 0.05)	0.04 (± 0.05)	0.04 (± 0.05)
t-STE with k-NN	0.09 (± 0.07)	0.10 (± 0.07)	0.06 (± 0.06)	0.08 (± 0.07)
t-STE with SVM linear	0.12 (± 0.09)	0.15 (± 0.11)	0.11 (± 0.09)	0.13 (± 0.09)
t-STE with SVM Gauss	0.09 (± 0.08)	0.08 (± 0.08)	0.07 (± 0.06)	0.08 (± 0.08)

and $0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100, 500, 1000$, respectively. When using the SVM algorithm with the Gaussian kernel, we chose the kernel bandwidth σ by means of 10-fold cross-validation from $0.01, 0.05, 0.1, 0.5, 1, 5$. The parameter k for Algorithm 4 was chosen from $1, 2, 3, 5, 7, 10, 15$ by means of a non-exhaustive variant of leave-one-out cross-validation as explained in Section 4.5.1. Clearly, the ordinal embedding approach outperforms Algorithms 3 and 4. However, one should judge the performance of our algorithms with regards to their great simplicity compared to the embedding approach. In doing so, we consider the 0-1 loss incurred by Algorithms 3 or 4 to be acceptable. As one might expect, working with T_All or T_All_reduced leads to a slightly lower

Table 4.4: Average purity (± standard deviation) of clusterings produced by the various methods when clustering the 56 cars from the classes of ordinary cars, sports cars, and off-road/sport utility vehicles into three clusters (average over 100 runs).

	All	All_reduced	T_All	T_All_reduced
Alg. 5 w., $k = 5$, $\sigma = 0.5$	0.82	0.82	0.86	0.88
Alg. 5 w., $k = 5$, $\sigma = 3$	0.84	0.82	0.86	0.86
Alg. 5 w., $k = 10$, $\sigma = 0.5$	0.91	0.84	0.95	0.93
Alg. 5 w., $k = 10$, $\sigma = 3$	0.84	0.91	0.86	0.89
Alg. 5 unw., $k = 5$	0.84	0.82	0.84	0.88
Alg. 5 unw., $k = 10$	0.84	0.84	0.86	0.89
GNMDS, $k = 5$, $\sigma = 0.5$	0.79 (± 0.12)	0.83 (± 0.12)	0.78 (± 0.15)	0.80 (± 0.15)
GNMDS, $k = 5$, $\sigma = 3$	0.78 (± 0.12)	0.84 (± 0.11)	0.76 (± 0.14)	0.83 (± 0.15)
GNMDS, $k = 10$, $\sigma = 0.5$	0.83 (± 0.11)	0.92 (± 0.03)	0.93 (± 0.04)	0.89 (± 0.13)
GNMDS, $k = 10$, $\sigma = 3$	0.78 (± 0.12)	0.88 (± 0.09)	0.92 (± 0.06)	0.95 (± 0.03)
SOE, $k = 5$, $\sigma = 0.5$	0.87 (± 0.01)	0.87 (± 0.04)	0.82 (± 0.08)	0.79 (± 0.10)
SOE, $k = 5$, $\sigma = 3$	0.82 (± 0.09)	0.83 (± 0.11)	0.75 (± 0.10)	0.73 (± 0.10)
SOE, $k = 10$, $\sigma = 0.5$	0.90 (± 0.04)	0.90 (± 0.03)	0.91 (± 0.04)	0.90 (± 0.03)
SOE, $k = 10$, $\sigma = 3$	0.93 (± 0.03)	0.93 (± 0.03)	0.91 (± 0.05)	0.89 (± 0.08)
STE, $k = 5$, $\sigma = 0.5$	0.75 (± 0.11)	0.73 (± 0.11)	0.74 (± 0.10)	0.73 (± 0.11)
STE, $k = 5$, $\sigma = 3$	0.75 (± 0.12)	0.76 (± 0.11)	0.74 (± 0.10)	0.76 (± 0.11)
STE, $k = 10$, $\sigma = 0.5$	0.90 (± 0.04)	0.87 (± 0.01)	0.90 (± 0.03)	0.88 (± 0.01)
STE, $k = 10$, $\sigma = 3$	0.90 (± 0.04)	0.87 (± 0.01)	0.77 (± 0.14)	0.88 (± 0.01)
t-STE, $k = 5$, $\sigma = 0.5$	0.87 (± 0.02)	0.87 (± 0.02)	0.86 (± 0.04)	0.88 (± 0.05)
t-STE, $k = 5$, $\sigma = 3$	0.85 (± 0.08)	0.89 (± 0.03)	0.76 (± 0.12)	0.79 (± 0.12)
t-STE, $k = 10$, $\sigma = 0.5$	0.92 (± 0.03)	0.92 (± 0.03)	0.92 (± 0.04)	0.91 (± 0.04)
t-STE, $k = 10$, $\sigma = 3$	0.94 (± 0.03)	0.94 (± 0.02)	0.92 (± 0.04)	0.91 (± 0.06)

misclassification rate than working with All or All_reduced.

Clustering

Like in the previous section on classification we removed the four outliers from the car data set. We then used Algorithm 5 and an ordinal embedding approach for clustering the remaining 56 cars into three clusters, aiming to recover the cars' grouping into classes of ordinary cars, sports cars, and off-road/sport utility vehicles. In the embedding approach we applied spectral clustering to a symmetric k-NN graph with Gaussian edge weights on an ordinal embedding of the data set as we did in Section 4.5.1. Table 4.4 shows the average purity (see equation (4.19) for its definition) of the clusterings produced by the considered methods with respect to our assumed ground truth partitioning. The average is over 100 runs of the experiment. Note that clusterings produced by Algorithm 5 and obtained in different runs only differ due to random effects in the K-means step of spectral clustering, while the clusterings produced by the embedding approach also differ due to the random initialization in the embedding methods. For this reason, standard deviations of the purity values achieved by the embedding approach are much larger than those of the purity values achieved by Algorithm 5 (which are on the order of machine epsilon) and are shown in Table 4.4 too. All methods perform nearly equally well, with the unweighted version of Algorithm 5 slightly inferior compared to the other methods when their parameters are chosen optimally. At least for Algorithm 5 working with the statements in T_All or T_All_reduced yields better results than working with the statements in All or All_reduced, but this does not seem to be the case for the ordinal embedding approach.

4.6 Discussion

In this chapter we have proposed algorithms for the problems of medoid estimation, outlier identification, classification, and clustering when given only a collection of statements of the kind ($\star$). We have shown that ordinal distance information of this type is intimately related to the lens depth function and the k-relative neighborhood graph. These relationships have not been discussed in the machine learning literature before. They allow us to make use of existing approaches to the considered problems based on depth functions and the k-RNG. Our algorithms are direct methods, that is they do not construct an ordinal embedding of the data set as an intermediate step. Hence, they avoid some of the drawbacks of an ordinal embedding approach that we discussed at the beginning of this chapter. In particular, our algorithms are deterministic and do not require to choose a dimension for the space of an embedding. Most important, as we have seen in the experiments of Section 4.5.1, our algorithms run faster by several orders of magnitude compared to an embedding approach, even without making use of their potential of simple and highly efficient parallelization. We believe that this makes our algorithms an useful alternative to the embedding approach and that they are applicable in situations in which ordinal embedding algorithms are not.

Our work inspires two main follow-up questions, which we discuss separately:

- A more local point of view: The problems studied in this chapter are global problems in the sense that they look at a data set as a whole. In contrast, local problems like density estimation or nearest neighbor search look at single data points and their neighborhoods with respect to the dissimilarity function ι, thus spotting only fragments of the data set. The tools used in this chapter, the lens depth function and the k-RNG, are global in their nature too. Indeed, as we have seen in Section 4.5.1, the lens depth function cannot detect outliers sitting in-between several modes of a data set since such outliers are globally seen at the heart of the data.

  It is interesting to consider local problems in a setting of ordinal distance information. A concept that becomes attractive then is that of local depth functions: Agostinelli and Romanazzi (2008, 2011) introduced a notion of localized simplicial depth, which can easily be transferred to the lens depth function and is then given by

$$LD_{\text{local}}(x;\tau,P) = \mathit{Probability}(x \in \mathit{Lens}(X,Y) \wedge \iota(X,Y) \leq \tau), \quad x \in \mathcal{X}, \qquad (4.20)$$

where X and Y are independent $\mathcal{X}$-valued random variables distributed according to a probability distribution P and $\tau > 0$ is a parameter. Agostinelli and Romanazzi have shown (theoretically for one-dimensional and empirically for multidimensional Euclidean data) that for τ tending to zero their local version of simplicial depth is closely related to the density function of the underlying distribution and that maximizing the local simplicial depth function provides reasonable estimates of the distribution's modes. We believe that such a connection also holds for the local lens depth function (4.20)—note that in one dimension the lens depth function coincides with the simplicial depth function. Unfortunately, unlike for the ordinary lens depth function, the local lens depth function cannot be evaluated with respect to an empirical distribution of a data set $\mathcal{D}$ given only ordinal distance information of the kind ($\star$) for $\mathcal{D}$. Even if we replace the event "$\iota(X,Y) \leq \tau$" by the event "$\iota(X,Y)$ is among the smallest τ distances between data points in $\mathcal{D}$", it is not clear at all how to evaluate or estimate (4.20). One solution would be to allow for additional ordinal distance information of the general type (1.1) like Ukkonen et al. (2015) do when studying the problem of density estimation based on statements of the kind ($\boxplus$), but this seems to be a rather unattractive way out. We have tried several heuristics for approximately evaluating a general comparison (1.1) given only statements of the kind ($\star$), like

$$\mathit{Probability}(x \in \mathit{Lens}(X,Y) \mid y \in \mathit{Lens}(X,Y)) \approx f(\iota(x,y))$$

for a monotonically decreasing function $f : \mathbb{R}_0^+ \rightarrow [0,1]$, which would be useful since we can easily estimate the probability on the left side. However, none of them was promising. They all suffer from the same problem, namely that the number of data points in $\mathit{Lens}(X,Y)$ can be small for two completely different reasons: either $\iota(X,Y)$ is small, or $\iota(X,Y)$ is large, but $\mathit{Lens}(X,Y)$ is located in an area of low probability. Unfortunately, there is no obvious way for distinguishing between these two reasons.

This raises the question whether density estimation or solving any other local problem is possible at all given only ordinal distance information of the kind ($\star$). Indeed, the answer is negative for intrinsically one-dimensional data sets: Consider data points $x_1, \ldots, x_n$ on the real line and assume ι to be the Euclidean metric. Then the ordinal distance information consisting of all statements of the kind ($\star$) only depends on the order of the data points: given any three data points, the most central one is always given by the data point sitting in the middle, and any order-preserving transformation of the data points will give rise to exactly the same ordinal distance information. For this reason it is impossible to estimate any local property of an underlying distribution, and ordinal distance information of the kind ($\star$) comes along with a substantial loss in information content compared to similarity triplets, that is answers to dissimilarity comparisons (1.2): while, under some assumptions on the data points, all similarity triplets asymptotically uniquely determine the actual positions of the points on the real line up to a similarity transformation (compare with Proposition 2.7), all statements of the kind ($\star$) only determine the ranking of the data points up to inversion. However, such a loss in information content does not seem to occur when dealing with Euclidean data sets of higher intrinsic dimension. In this case we conjecture the uniqueness property as discussed in Section 1.4 to hold for ordinal embedding based on statements of the kind ($\star$). We formulate the conjecture analogously to Theorem 2.3 and use the characterization (4.1) of a statement of the kind ($\star$).

Conjecture 4.1 (Uniqueness property for ordinal embedding based on statements of the kind ($\star$))**.** *Let $d \geq 2$ and $K = B_r(z) \subseteq \mathbb{R}^d$ be a closed and bounded ball (for some arbitrary $r > 0$, $z \in \mathbb{R}^d$). Let $(x_n)_{n\in\mathbb{N}}$ be a sequence of points $x_n \in K$ such that $\{x_n : n \in \mathbb{N}\}$ is dense in K. Let $0 < R < \infty$ and $(\varphi_n)_{n\in\mathbb{N}}$ be a sequence of functions $\varphi_n : \{x_1, \ldots, x_n\} \to U_R(0) \subseteq \mathbb{R}^d$ with the property that for all $n \in \mathbb{N}$ and for all $i, j, k \in \{1, \ldots, n\}$,*

$$\begin{gathered}\big(\|x_i - x_j\| < \|x_j - x_k\|\big) \;\wedge\; \big(\|x_i - x_k\| < \|x_j - x_k\|\big) \;\Rightarrow \\ \big(\|\varphi_n(x_i) - \varphi_n(x_j)\| < \|\varphi_n(x_j) - \varphi_n(x_k)\|\big) \wedge \big(\|\varphi_n(x_i) - \varphi_n(x_k)\| < \|\varphi_n(x_j) - \varphi_n(x_k)\|\big).\end{gathered}$$

Then there exists a sequence $(S_n)_{n\in\mathbb{N}}$ of similarity transformations $S_n : \mathbb{R}^d \to \mathbb{R}^d$ such that

$$\|S_n - \varphi_n\|_{\infty(\{x_1,\ldots,x_n\})} \to 0 \quad \text{as} \quad n \to \infty.$$

Support for this conjecture comes from a large number of experiments similar to the one shown in Figure 1.3. Hence, there is hope: if Conjecture 4.1 holds, when dealing with Euclidean data sets of intrinsic dimension greater than one, in principle it should be possible to solve any local problem that is solvable in a "standard" setting of cardinal distance information also in a setting of ordinal distance information of the kind ($\star$). However, it remains an open problem how to solve a local problem in practice except for an embedding approach.

- Active setting: Algorithms 1 to 5 are designed for a batch setting, that is they can deal with arbitrary collections of statements of the kind ($\star$) that are gathered before the application of the algorithm and are provided as input all at once. We might also be interested in algorithms in an active setting, in which we can actively query statements for intentionally chosen triples of objects. In such a scenario an algorithm for a machine learning task should interact with the process of querying statements and adaptively choose triples of objects for which statements are to be queried in such a way that the task at hand is solved as fast, accurately, cheaply, ... as possible. For the problems of medoid estimation or outlier identification it is easy to modify Algorithm 1 and Algorithm 2 in order to derive adaptive versions: Starting with rough estimates of values $LD(O)$ for every object O in the data set $\mathcal{D}$, one could immediately rule out some objects with very small (or high) estimated values. Subsequently, only the estimates of the values of the remaining objects are improved by querying further statements only for them. This strategy has been suggested by Heikinheimo and Ukkonen (2013) for their method for medoid estimation. It is interesting whether such a strategy comes with any guarantees, whether there might be better alternatives (of course, this depends on what one wants to achieve), or whether similar approaches apply to Algorithms 3 to 5.

Chapter 5

Kernel functions based on similarity triplets

We have argued at the beginning of Chapter 4 that the main approach to machine learning problems based on ordinal distance information, which consists of constructing an ordinal embedding of the data set and solving the problem on the embedding, has a number of drawbacks. We suggested to aim at solving problems directly, that is without constructing an ordinal embedding as an intermediate step. The main contribution of Chapter 4 is that we then proposed algorithms that meet this requirement. Each of these algorithms is designed for a specific machine learning problem.

In this chapter, we follow up on the idea of solving problems based on ordinal data directly. We propose two ways of defining a data-dependent kernel function on a data set when given only an arbitrary collection of similarity triplets (compare with Section 1.3). Such a kernel function can subsequently be used to apply any kernel method to the data set. Hence, our proposed kernel functions provide a generic alternative to the ordinal embedding approach, that is they can be used to solve a variety of problems. Like the algorithms of Chapter 4, the methods presented in this chapter are appealingly simple and avoid some of the drawbacks of an ordinal embedding approach. In particular, we observe our kernel functions to run significantly faster than well-known ordinal embedding algorithms.

5.1 Setup and notation for Chapter 5

We deal with a data set $\mathcal{D} = \{x_1, \ldots, x_n\}$ comprising n indexed objects and a collection $\mathcal{S}$ of similarity triplets for $\mathcal{D}$. Recall from Section 1.3 that similarity triplets are answers to dissimilarity comparisons of the restricted form (1.2), that is

$$\iota(A, B) \overset{?}{<} \iota(A, C).$$

To simplify presentation, we assume that for all triples of distinct objects $x_i, x_j, x_k \in \mathcal{D}$ either $\iota(x_i, x_j) < \iota(x_i, x_k)$ or $\iota(x_i, x_j) > \iota(x_i, x_k)$ is true. However, we allow $\mathcal{S}$ to contain incorrect similarity triplets. We assume similarity triplets in $\mathcal{S}$ to be encoded by ordered triples: an ordered triple of distinct objects $(x_i, x_j, x_k) \in \mathcal{S}$ is interpreted as $\iota(x_i, x_j) < \iota(x_i, x_k)$. We refer to x_i as the anchor object in the similarity triplet (x_i, x_j, x_k).

Our proposed kernel functions can deal with an arbitrary collection $\mathcal{S}$ of similarity triplets, and we do not make any assumptions on how $\mathcal{S}$ is related to the set of all similarity triplets for $\mathcal{D}$, that is the set of answers to all possible dissimilarity comparisons (1.2). In Section 5.2.6, however, we will discuss a strategy for choosing comparisons that should be evaluated for creating $\mathcal{S}$ in case one can choose them. This strategy drastically increases the meaningfulness of our kernel functions relative to the size of $\mathcal{S}$.

5.2 Our kernel functions

We present two ways of defining a data-dependent kernel function on $\mathcal{D}$ when only given a collection $\mathcal{S}$ of similarity triplets for $\mathcal{D}$. Our proposed kernel functions measure similarity between two objects in $\mathcal{D}$ by comparing to which extent the two objects give rise to resembling similarity triplets. The hope is that this quantifies the relative difference in the locations of the two objects in $\mathcal{D}$. We provide a geometric interpretation of our kernel functions that supports this hope in Section 5.2.5. Our experiments in Section 5.2.5 and Section 5.3 show that the similarity scores defined by our kernel functions are meaningful for a range of both artificial and real data sets.

For the moment assume that contradicting triples (x_i, x_j, x_k) and (x_i, x_k, x_j) cannot be present in $\mathcal{S}$ at the same time. We will discuss how to deal with the general case in Section 5.2.3.

5.2.1 Kernel function k_1

Our first kernel function is based on the following idea: We fix two objects x_a and x_b of $\mathcal{D}$. In order to compute a similarity score between x_a and x_b we would like to rank all objects in $\mathcal{D}$ with respect to their distance from x_a and also rank them with respect to their distance from x_b, and take a similarity score between these two rankings as similarity score between x_a and x_b. One possibility to measure similarity between rankings is given by the famous Kendall tau correlation coefficient (Kendall, 1938), which is also known as Kendall's $\boldsymbol{\tau}$: for two rankings of n items, Kendall's $\boldsymbol{\tau}$ between the two rankings is the fraction of concordant pairs of items minus the fraction of discordant pairs of items. Here, a pair of two items i_1 and i_2 is concordant if $i_1 \prec i_2$ or $i_1 \succ i_2$ according to both rankings, and discordant if it satisfies $i_1 \prec i_2$ according to one and $i_1 \succ i_2$ according to the other ranking. Formally, a ranking is represented by a permutation $\sigma : \{1, \ldots, n\} \to \{1, \ldots, n\}$ such that $\sigma(i) \neq \sigma(j)$, $i \neq j$, and $\sigma(i) = m$ means that item i is ranked at the m-th position. Given two rankings σ_1 and σ_2, the number of concordant pairs equals

$$\begin{aligned} f_c(\sigma_1, \sigma_2) = \sum_{i<j} \big[& \mathbb{1}\{\sigma_1(i) < \sigma_1(j)\} \mathbb{1}\{\sigma_2(i) < \sigma_2(j)\} \\ & + \mathbb{1}\{\sigma_1(i) > \sigma_1(j)\} \mathbb{1}\{\sigma_2(i) > \sigma_2(j)\} \big], \end{aligned}$$

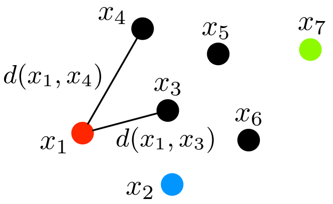


Figure 5.1: Illustration of the idea behind our kernel function k_1. In order to compute a similarity score between x_1 (in red) and x_2 (in blue) we would like to rank all objects with respect to their distance from x_1 and also with respect to their distance from x_2 and compute Kendall's $\boldsymbol{\tau}$ between the two rankings. In this example, the objects would rank as $x_1 \prec x_3 \prec x_2 \prec x_4 \prec x_5 \prec x_6 \prec x_7$ and $x_2 \prec x_3 \prec x_6 \prec x_1 \prec x_5 \prec x_4 \prec x_7$, respectively. Kendall's $\boldsymbol{\tau}$ between these two rankings is 1/3, and this would be the similarity score between x_1 and x_2. For comparison, the similarity score between x_1 and x_7 (in green) would be $-5/7$, and between x_2 and x_7 it would be $-3/7$.

the number of discordant pairs equals

$$f_d(\sigma_1, \sigma_2) = \sum_{i<j} \Big[\mathbb{1}\{\sigma_1(i) < \sigma_1(j)\}\mathbb{1}\{\sigma_2(i) > \sigma_2(j)\} \\ + \mathbb{1}\{\sigma_1(i) > \sigma_1(j)\}\mathbb{1}\{\sigma_2(i) < \sigma_2(j)\}\Big],$$

and Kendall's $\boldsymbol{\tau}$ between σ_1 and σ_2 is given by

$$\boldsymbol{\tau}(\sigma_1, \sigma_2) = \frac{f_c(\sigma_1, \sigma_2) - f_d(\sigma_1, \sigma_2)}{\binom{n}{2}}.$$

It has been established only recently that Kendall's $\boldsymbol{\tau}$ is actually a kernel function on the set of total rankings (Jiao and Vert, 2015). Consequently, by measuring similarity between the two rankings of objects (one with respect to their distance from x_a and one with respect to their distance from x_b) with Kendall's $\boldsymbol{\tau}$ we would not only compute a similarity score between x_a and x_b, but would even end up with a kernel function on $\mathcal{D}$ since the following holds: for any mapping $h : \mathcal{D} \rightarrow \mathcal{Z}$ and kernel function $k : \mathcal{Z} \times \mathcal{Z} \rightarrow \mathbb{R}$, the composition $k \circ (h, h) : \mathcal{D} \times \mathcal{D} \rightarrow \mathbb{R}$ is a kernel function. This idea is illustrated with an example of a data set comprising seven points in the Euclidean plane in Figure 5.1.

In our situation, the problem is that in most cases $\mathcal{S}$ will contain only a small fraction of all similarity triplets and also that some of the triplets in $\mathcal{S}$ may be incorrect. This will not allow us to rank all objects with respect to their distance from any fixed object based on the similarity triplets in $\mathcal{S}$. We therefore have to adapt the procedure. For doing so we consider a feature map that corresponds to the kernel function k_τ that we just described. By a feature map corresponding to a kernel function $k : \mathcal{D} \times \mathcal{D} \rightarrow \mathbb{R}$ we mean a mapping $\Phi : \mathcal{D} \rightarrow \mathbb{R}^d$ for some $d \in \mathbb{N}$ such that

$$k(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle_{\mathbb{R}^d} = \Phi(x_i)^T \cdot \Phi(x_j).$$

It is easy to see from the above formulas (also compare with Jiao and Vert, 2015) that a feature map corresponding to the described kernel function k_τ is given by $\Phi_{k_\tau} : \mathcal{D} \rightarrow \mathbb{R}^{\binom{n}{2}}$

with

$$\Phi_{k_\tau}(x_a) = \frac{1}{\sqrt{\binom{n}{2}}} \cdot \Big(\mathbb{1}\{\iota(x_a, x_i) < \iota(x_a, x_j)\} - \mathbb{1}\{\iota(x_a, x_i) > \iota(x_a, x_j)\}\Big)_{1 \leq i < j \leq n}.$$

In our situation, where we are only given $\mathcal{S}$ and will not be able to evaluate Φ_{k_τ} in most cases, we have to replace Φ_{k_τ} by an approximation: up to a normalizing factor, we simply replace an entry in $\Phi_{k_\tau}(x_a)$ by zero if we cannot evaluate it based on the triplets in $\mathcal{S}$. More precisely, we consider the feature map $\Phi_{k_1} : \mathcal{D} \to \mathbb{R}^{\binom{n}{2}}$ given by

$$\begin{aligned}\Phi_{k_1}(x_a) = {} & \frac{1}{\sqrt{|\{(x_i, x_j, x_k) \in \mathcal{S} : x_i = x_a\}|}} \cdot \\ & \Big(\mathbb{1}\{(x_a, x_i, x_j) \in \mathcal{S}\} - \mathbb{1}\{(x_a, x_j, x_i) \in \mathcal{S}\}\Big)_{1 \leq i < j \leq n}\end{aligned} \tag{5.1}$$

and define our first proposed kernel function $k_1 : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$ by

$$k_1(x_i, x_j) = \Phi_{k_1}(x_i)^T \cdot \Phi_{k_1}(x_j). \tag{5.2}$$

Note that the scaling factor in the definition of Φ_{k_1}, ensuring that the feature embedding lies on the unit sphere, is crucial whenever the number of similarity triplets in which an object appears as anchor object is not approximately constant over the different objects. For ease of exposition we have assumed that every object in $\mathcal{D}$ appears at least once as an anchor object in a similarity triplet in $\mathcal{S}$. In the unlikely case that x_a does not appear at least once as an anchor object, meaning that we do not have any information for ranking the objects in $\mathcal{D}$ with respect to their distance from x_a at all, we simply set $\Phi_{k_1}(x_a)$ to zero, which is consistent with (5.1) under the convention "0/0=0".

5.2.2 Kernel function k_2

Our second kernel function is based on a similar idea, however, now we do not consider x_a and x_b as anchor objects when measuring their similarity, but rather compare whether they rank similarly with respect to their distances from the various other objects. Concretely, up to normalization, we would like to count the number of pairs of objects $(x_i, x_j) \in \mathcal{D} \times \mathcal{D}$ for which the comparisons

$$\iota(x_i, x_a) \overset{?}{<} \iota(x_i, x_j) \quad \text{and} \quad \iota(x_i, x_b) \overset{?}{<} \iota(x_i, x_j) \tag{5.3}$$

yield the same result and subtract the number of pairs for which these comparisons yield different results. See Figure 5.2 for an illustration of this idea.

Adapted to our situation of being only given $\mathcal{S}$ it corresponds to considering the feature map $\Phi_{k_2} : \mathcal{D} \to \mathbb{R}^{n^2}$ given by

$$\begin{aligned}\Phi_{k_2}(x_a) = {} & \frac{1}{\sqrt{|\{(x_i, x_j, x_k) \in \mathcal{S} : x_j = x_a \vee x_k = x_a\}|}} \cdot \\ & \Big(\mathbb{1}\{(x_i, x_a, x_j) \in \mathcal{S}\} - \mathbb{1}\{(x_i, x_j, x_a) \in \mathcal{S}\}\Big)_{1 \leq i, j \leq n}\end{aligned} \tag{5.4}$$

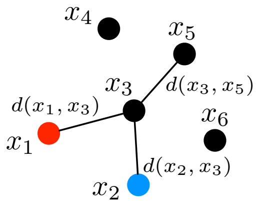


Figure 5.2: Illustration of the idea behind our kernel function k_2. In order to compute a similarity score between x_1 (in red) and x_2 (in blue) we would like to check for every pair of objects (x_i, x_j) whether the distance comparisons $\iota(x_i, x_1) \overset{?}{<} \iota(x_i, x_j)$ and $\iota(x_i, x_2) \overset{?}{<} \iota(x_i, x_j)$ yield the same result or not. Here, we have 32 pairs for which they yield the same result (e.g., (x_3, x_7) is one such a pair) and 17 pairs for which they do not (e.g., (x_3, x_5)). We would assign $7^{-2} \cdot (32 - 17) = 15/49$ as similarity score between x_1 and x_2. The similarity score between x_1 and x_7 (in green) would be 3/49, and between x_2 and x_7 it would be 1/49.

and defining our second proposed kernel function $k_2 : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$ by

$$k_2(x_i, x_j) = \Phi_{k_2}(x_i)^T \cdot \Phi_{k_2}(x_j). \tag{5.5}$$

The scaling factor in the definition of Φ_{k_2} is crucial whenever there are objects that appear (not as anchor object) in more similarity triplets than others. Again, we apply the convention "0/0=0" whenever we encounter a denominator equaling zero in (5.4).

5.2.3 Contradicting similarity triplets and combining k_1 with k_2

If $\mathcal{S}$ contains contradicting triples (x_i, x_j, x_k) and (x_i, x_k, x_j) and there might be triples that are present repeatedly, we can alter the definition of Φ_{k_1} or Φ_{k_2} as follows: if $\#\{(x_a, x_i, x_j) \in \mathcal{S}\}$ denotes the number of how often the triple (x_a, x_i, x_j) appears in $\mathcal{S}$, we set

$$\Phi_{k_1}(x_a) = \frac{\widetilde{\Phi}_{k_1}(x_a)}{\left\|\widetilde{\Phi}_{k_1}(x_a)\right\|},$$

where

$$\widetilde{\Phi}_{k_1}(x_a) = \left(\frac{\#\{(x_a, x_i, x_j) \in \mathcal{S}\} - \#\{(x_a, x_j, x_i) \in \mathcal{S}\}}{\#\{(x_a, x_i, x_j) \in \mathcal{S}\} + \#\{(x_a, x_j, x_i) \in \mathcal{S}\}} \right)_{1 \leq i < j \leq n}.$$

The definition of Φ_{k_2} can be revised in an analogous way. In doing so, we incorporate a simple estimate of the likelihood of a triple being correct.

We can combine k_1 with k_2 in order to obtain another kernel functions: for parameters $\mu_1, \mu_2 \geq 0$ we define $k_3^{\mu_1,\mu_2} : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$ as

$$k_3^{\mu_1,\mu_2} = \mu_1 k_1 + \mu_2 k_2.$$

A corresponding feature map is given by $\Phi_{k_3^{\mu_1,\mu_2}} : \mathcal{D} \to \mathbb{R}^{\binom{n}{2}+n^2}$ with

$$\Phi_{k_3^{\mu_1,\mu_2}}(x_a) = \begin{pmatrix} \sqrt{\mu_1}\ \Phi_{k_1}(x_a) \\ \sqrt{\mu_2}\ \Phi_{k_2}(x_a) \end{pmatrix}.$$

There are further possibilities for building up new kernel functions from existing ones. For example, one could also consider the kernel functions $k_1 \cdot k_2$ or $\exp(k_i)$, $i = 1, 2$ (e.g., Hofmann et al., 2008).

5.2.4 Reducing diagonal dominance

If the number $|\mathcal{S}|$ of given similarity triplets is small, our kernel functions suffer from a problem that is shared by many other kernel functions defined on complex data: the feature maps Φ_{k_1} and Φ_{k_2} map the objects in $\mathcal{D}$ to sparse vectors, that is almost all of their entries are zero. As a consequence, two different feature vectors $\Phi_{k_i}(x_a)$ and $\Phi_{k_i}(x_b)$ appear to be almost orthogonal and the similarity score $k_i(x_a, x_b)$ is much smaller than the self-similarity scores $k_i(x_a, x_a)$ or $k_i(x_b, x_b)$. This phenomenon, usually referred to as diagonal dominance of the kernel function, has been observed to pose difficulties for the kernel methods using the kernel function, and several ways have been proposed for dealing with it (Schölkopf et al., 2002, Greene and Cunningham, 2006). In all our experiments we deal with diagonal dominance in the following simple way: Let k denote a kernel function and K the kernel matrix on $\mathcal{D}$, that is $K = (k(x_i, x_j))_{i,j=1}^n$, which would be the input to a kernel method. Then we replace K by $K - \lambda_{\min} I$, where $I \in \mathbb{R}^{n \times n}$ denotes the identity matrix and $\lambda_{\min}$ is the smallest eigenvalue of K. Note that $\lambda_{\min} \geq 0$ and that it is the largest number that we can subtract from the diagonal of K such that the resulting matrix is still positive semi-definite.

5.2.5 Geometric intuition

Intuitively, our kernel functions measure similarity between x_a and x_b by quantifying to which extent x_a and x_b can be expected to be located in the same region of $\mathcal{D}$: Think of $\mathcal{D}$ as being a subset of $\mathbb{R}^d$ and ι being the Euclidean metric. A similarity triplet $\iota(x_a, x_i) < \iota(x_a, x_j)$ then tells us that x_a resides in the halfspace defined by the hyperplane that is perpendicular to the line segment connecting x_i and x_j and goes through the segment's midpoint. If there is also a similarity triplet $\iota(x_b, x_i) < \iota(x_b, x_j)$, x_a and x_b thus are located in the same halfspace (assuming the correctness of the similarity triplets), and this is reflected by a higher value of $k_1(x_a, x_b)$. Similarly, a similarity triplet $\iota(x_i, x_a) < \iota(x_i, x_j)$ tells us that x_a is located in a ball with radius $\iota(x_i, x_j)$ centered at x_i, and the value of $k_2(x_a, x_b)$ is higher if there is a similarity triplet $\iota(x_i, x_b) < \iota(x_i, x_j)$ telling us that x_b is located in this ball too and it is smaller if there is a similarity triplet $\iota(x_i, x_j) < \iota(x_i, x_b)$ telling us that x_b is not located in this ball.

The similarity scores between x_a and x_b defined by k_1 and k_2 do not only depend on $\iota(x_a, x_b)$, but rather on the locations of x_a and x_b within $\mathcal{D}$ and on how the points in $\mathcal{D}$ are spread in the space since this affects how the various hyperplanes or balls are related to each other. Consider the example illustrated in Figure 5.3: Let $\iota(x_3, x_n) = 1$ implying that $\iota(x_i, x_{i+1}) = \Theta(1/n)$, $3 \leq i < n$, and $\iota(x_1, x_2) > \iota(x_2, x_n) > \iota(x_1, x_n) > \iota(x_2, x_3) > \iota(x_1, x_3) > 1$ be arbitrarily large. Although x_1 and x_2 are located at the maximal distance to each other, they satisfy $\iota(x_1, x_i) < \iota(x_1, x_j)$ and $\iota(x_2, x_i) < \iota(x_2, x_j)$ for all $3 \leq i < j \leq n$, and hence both x_1 and x_2 are jointly located in all the halfspaces obtained from these similarity triplets. Note that these halfspaces can be arranged as a sequence of increasing subsets. It is easy to see that we end up with $k_1(x_1, x_2) \to 1$ as $n \to \infty$,

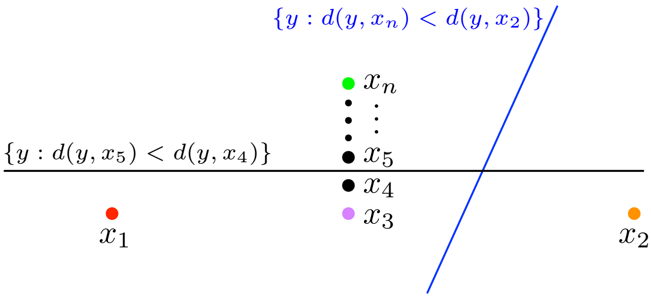

$\{y : d(y, x_n) < d(y, x_2)\}$
x_n
$\{y : d(y, x_5) < d(y, x_4)\}$
x_5
x_4
x_3
x_1
x_2

Figure 5.3: The kernel function k_1 measures similarity between two objects essentially by counting in how many of the halfspaces that are obtained from distance comparisons the two objects reside at the same time. The outcome does not only depend on the distance between the two objects, but also on their location within the data set: although x_1 and x_2 are located far apart from each other, the kernel function k_1 considers them to be very similar. See the running text for details.

assuming k_1 is computed based on all similarity triplets, all of which are correct. On the other hand, the distance between x_3 and x_n is much smaller, but there are many points in between them and the hyperplanes obtained from the distance comparisons (1.2) with these points separate x_3 and x_n. We end up with $k_1(x_3, x_n) \to -1$ as $n \to \infty$. Depending on the task at hand, this may be desirable or not.

Let us examine the meaningfulness of our kernel functions by calculating them on four visualizable data sets. The first three data sets consist of 400 points in $\mathbb{R}^2$ and there ι equals the Euclidean metric. The fourth one consists of the vertices of an undirected graph from a stochastic block model and there ι equals the shortest path distance. We computed the kernel functions k_1, k_2, and $k_3^{1,1}$ based on 10% of all similarity triplets (chosen uniformly at random without replacement from the set of all similarity triplets), all of which were correct. The results are shown in Figure 5.4. The first plot of a row shows the data set. The second plot shows the negated distance matrix on the data set. Next, we can see the kernel matrices. The last plot of a row shows the similarity scores (encoded by color) based on k_1 between one fixed point (shown as a black cross) and the other points in the data set. Clearly, the kernel matrices reflect the block structures of the distance matrices. Also, the similarity scores are the smaller the larger the distances from the fixed points are. A situation like in the example of Figure 5.3 does not occur.

5.2.6 Landmark design

Our kernel functions are designed as to extract information from an arbitrary collection $\mathcal{S}$ of similarity triplets. However, by construction, a single similarity triplet is useless, what matters is the concurrent presence of two triplets: $k_1(x_a, x_b)$ is only affected by pairs of similarity triplets answering

$$\iota(x_a, x_i) \overset{?}{<} \iota(x_a, x_j) \quad \text{and} \quad \iota(x_b, x_i) \overset{?}{<} \iota(x_b, x_j),$$

while $k_2(x_a, x_b)$ is only affected by pairs of similarity triplets answering (5.3). Hence, when we can choose which dissimilarity comparisons of the form (1.2) are evaluated for creating $\mathcal{S}$ (e.g., in a crowdsourcing scenario), we should aim at maximizing the number

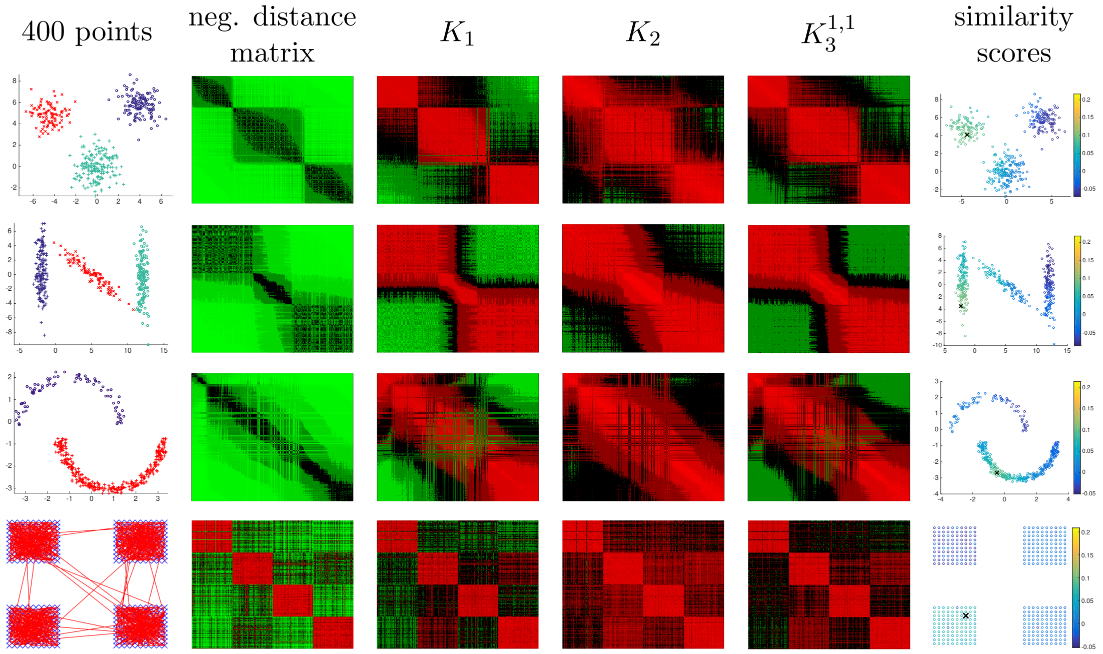


Figure 5.4: Kernel matrices for four data sets consisting of 400 points based on 10% of all similarity triplets. The first, the second, and the third data set consist of points in $\mathbb{R}^2$ and there ι equals the Euclidean metric. The fourth data set consists of the vertices of an undirected graph and there ι equals the shortest path distance. 1st plot: The data points. 2nd plot: The negated distance matrix. 3rd / 4th / 5th plot: The kernel matrix corresponding to k_1 / k_2 / $k_3^{1,1}$. 6th plot: Similarity scores (encoded by color) based on k_1 between a fixed point (shown as a black cross) and the other points.

of appropriate pairs of similarity triplets. This can easily be achieved by means of a landmark design inspired from so-called landmark multidimensional scaling (de Silva and Tenenbaum, 2004): for a small subset of landmark objects $\mathcal{L} \subseteq \mathcal{D}$ only comparisons of the form (when working with k_1)

$$\iota(x_i, x_j) \overset{?}{<} \iota(x_i, x_k)$$

or of the form (when working with k_2)

$$\iota(x_j, x_i) \overset{?}{<} \iota(x_j, x_k)$$

with $x_i \in \mathcal{D}$ and $x_j, x_k \in \mathcal{L}$ are evaluated. The landmark objects can be chosen either randomly or, if available, based on additional knowledge about $\mathcal{D}$ and the task at hand.

5.2.7 Computational complexity

General $\mathcal{S}$

A naive implementation of our kernel functions explicitly computes the feature vectors $\Phi_{k_1}(x_i)$ or $\Phi_{k_2}(x_i)$, $i = 1, \ldots, n$, and subsequently calculates the kernel matrix K by means of (5.2) or (5.5). In doing so, we store the feature vectors in the feature matrix $\Phi_{k_1}(\mathcal{D}) = (\Phi_{k_1}(x_i))_{i=1}^n \in \mathbb{R}^{\binom{n}{2} \times n}$ or $\Phi_{k_2}(\mathcal{D}) = (\Phi_{k_2}(x_i))_{i=1}^n \in \mathbb{R}^{n^2 \times n}$. Proceeding this way is straightforward and simple, requiring to go through $\mathcal{S}$ only once, but comes with a computational cost of $\mathcal{O}(|\mathcal{S}| + n^4)$ operations. Note that the number of different distance comparisons of the form (1.2) is $\mathcal{O}(n^3)$ and hence one might expect that $|\mathcal{S}| \in \mathcal{O}(n^3)$ and $\mathcal{O}(|\mathcal{S}| + n^4) = \mathcal{O}(n^4)$. By performing (5.2) or (5.5) in terms of matrix multiplication $\Phi_{k_1}(\mathcal{D})^T \cdot \Phi_{k_1}(\mathcal{D})$ or $\Phi_{k_2}(\mathcal{D})^T \cdot \Phi_{k_2}(\mathcal{D})$ and applying Strassen's algorithm (Higham, 1990) instead of standard matrix multiplication one can slightly reduce the number of operations to $\mathcal{O}(|\mathcal{S}| + n^{3.81})$, but still this is infeasible for any somewhat large data set. Currently, it is not clear to us whether it is really necessary to explicitly compute the feature vectors $\Phi_{k_1}(x_i)$ or $\Phi_{k_2}(x_i)$ or one can do better than the naive implementation. Nevertheless, as we will see in the experiments in Section 5.3, even with the naive implementation computing our kernel functions takes significantly less time than computing an ordinal embedding.

Landmark design

If we know that $\mathcal{S}$ contains only dissimilarity comparisons involving landmark objects as explained in Section 5.2.6, we can adapt the feature matrices such that $\Phi_{k_1}(\mathcal{D}) \in \mathbb{R}^{\binom{|\mathcal{L}|}{2} \times n}$ or $\Phi_{k_2}(\mathcal{D}) \in \mathbb{R}^{|\mathcal{L}|^2 \times n}$. In doing so, we reduce the number of required operations to $\mathcal{O}(|\mathcal{S}| + \min\{|\mathcal{L}|^2, n\}^{\log_2(7/8)} |\mathcal{L}|^2 n^2)$, which is $\mathcal{O}(|\mathcal{S}| + |\mathcal{L}|^{1.62} n^2)$ if $|\mathcal{L}|^2 \leq n$. Note that in this case the number of different possible distance comparisons is $\mathcal{O}(|\mathcal{L}|^2 n)$ and hence one might expect that $|\mathcal{S}| \in \mathcal{O}(|\mathcal{L}|^2 n)$.

In both cases, whenever the number of given similarity triplets $|\mathcal{S}|$ is small compared to the number of all different distance comparisons under consideration, the feature matrix $\Phi_{k_1}(\mathcal{D})$ or $\Phi_{k_2}(\mathcal{D})$ is sparse with only $\mathcal{O}(|\mathcal{S}|)$ non-zero entries and methods for sparse matrix multiplication decrease computational complexity (Gustavson, 1978, Kaplan et al., 2006).

5.3 Experiments

We performed several experiments in order to study the meaningfulness of our kernel functions and to make a performance comparison with an ordinal embedding approach. Our experiments confirm what we have already seen in Figure 5.4: our kernel functions are meaningful and can capture the structure of a data set when given sufficiently many (correct) similarity triplets. We find that, in general, they require a higher number of similarity triplets than an ordinal embedding approach, but in a landmark design our kernel functions can compete with an embedding approach regarding the required number of triplets. In any case, like our algorithms of Chapter 4, our kernel functions run significantly faster than ordinal embedding algorithms.

We first present experiments with synthetically generated similarity triplets, in which we systematically study the performance of our kernel functions in clustering tasks. After that we demonstrate the meaningfulness of our kernel functions by applying them to the car data set introduced in Section 4.5.2 with crowdsourced similarity triplets.

5.3.1 Synthetically generated similarity triplets

We studied our kernel functions when used in order to apply kernel K-means clustering (Dhillon et al., 2004) to subsets of USPS digits 1, 2, or 3 and compared our approach to an ordinal embedding approach. The ordinal embedding approach consists of clustering the data set through clustering an ordinal embedding of it. We tried the GNMDS (Agarwal et al., 2007), the CKL (Tamuz et al., 2011), and the t-STE (van der Maaten and Weinberger, 2012) algorithms in the Matlab implementation provided by van der Maaten and Weinberger (2012) for constructing an ordinal embedding, and we used the ordinary K-means algorithm (e.g., Shalev-Shwartz and Ben-David, 2014, Section 22.2) for clustering an embedding. We set all parameters of the embedding algorithms except the dimension of the space of the embedding to the provided default parameters as we did in the experiments of Section 4.5. The parameter μ of the CKL algorithm, which does not come with a default, was set to 0.1 since we observed good results with this value. Note that in the unsupervised clustering tasks that we are considering there is no immediate way of performing cross-validation for choosing parameters. We always provided the correct number of clusters, that is three, as input and set the number of replicates in K-means and kernel K-means to five and the maximum number of iterations to 100. For assessing the quality of a clustering we computed its purity (compare with Section 4.5.1 and see (4.19) for its definition) with respect to the ground truth class labels of the digits. For comparison, all plots showing purity values also show the purity of the clustering that was obtained by applying ordinary K-means to the original point set (which was, of course, not known to either our kernel functions or the embedding algorithms). They also show the purity of a random clustering in which data points were randomly assigned to one of three clusters independently of each other with probability 1/3. All computations were performed in Matlab R2016a on a MacBook Pro with 2.9 GHz Intel Core i7 and 8 GB 1600 MHz DDR3. In order to make a fair comparison of running times we did not use MEX files or sparse matrix operations in the implementation of our kernel functions. All plots show results averaged over 10 runs of an experiment.

We first considered the scenario of a general collection $\mathcal{S}$ of similarity triplets and then looked at a landmark design (compare with Section 5.2.6). In both cases we chose input similarity triplets uniformly at random without replacement from the set of answers to all possible distance comparisons under consideration, where the dissimilarity function ι always equaled the Euclidean metric and answers were incorrect with some error probability $0 \leq errorprob \leq 1$ independently of each other. This noise model is similar to Noise model I in the experiments of Section 4.5.1.

General $\mathcal{S}$

Figure 5.5 shows in the first row the purity of the clusterings produced by kernel K-means based on k_1, k_2, or $k_3^{1,1}$, and ordinary K-means applied to the output of the various ordinal embedding algorithms for 400 points chosen uniformly at random from USPS digits 1, 2, or 3. In the first and the second plot we can study the purity as a function of the number of input similarity triplets, in the third plot as a function of *errorprob*. For the embedding approach, the dimension of the space of the embedding was always set to five. This choice gave better results than a choice of two and similar results as a choice of

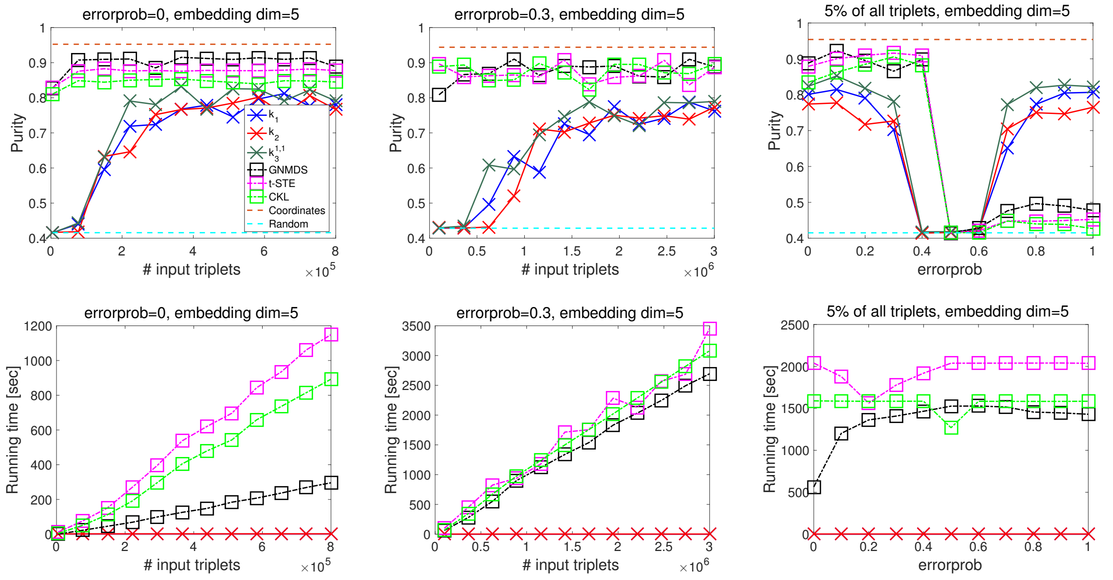


Figure 5.5: General $\mathcal{S}$ — 400 points from USPS digits 1, 2, or 3 with Euclidean metric. First row: Purity as a function of the number of input similarity triplets (1st & 2nd plot) and as a function of *errorprob* (3rd plot) for kernel K-means based on one of our kernel functions and for ordinary K-means applied to the ordinal embedding of one of the three embedding methods and to the original point set (brown line). The cyan line shows the purity of a random clustering as a lower baseline. Second row: Corresponding running times in seconds for computing k_1 and k_2 and the ordinal embeddings.

ten (plots omitted). The second row of Figure 5.5 shows the running time for computing our kernel functions or the ordinal embeddings corresponding to the plots in the first row. The ordinal embedding approach clearly outperforms our kernel functions in terms of the number of similarity triplets that are required for producing a reasonable result (1st row, 1st & 2nd plot). Our kernel functions are also more sensitive to noise in the similarity triplets (1st row, 3rd plot). Interestingly, our kernel functions yield high purity values for $errorprob \geq 0.7$. In hindsight, this is not surprising: if $errorprob = 1$ and thus every similarity triplet is incorrect, we simply end up with the feature map $-\Phi_k$, when $+\Phi_k$ is the feature map corresponding to one of our kernel functions based on only correct triplets, and hence with the same kernel function as for $errorprob = 0$. Clearly, our kernel functions are highly superior regarding running time (2nd row).

Landmark design

We studied the performance of our kernel functions in a landmark design, in which we consider only similarity triplets that are answers to special dissimilarity comparisons (1.2). The aim of a landmark design is to increase the meaningfulness of our kernel functions relative to the number of input triplets (compare with Section 5.2.6). We compared to the ordinal embedding approach in two scenarios: In one case, the embedding algorithms were provided the same similarity triplets as input as our kernel functions. In the other case, they were provided a same number of similarity triplets chosen uniformly at random with replacement from all triplets, that is answers to comparisons (1.2)

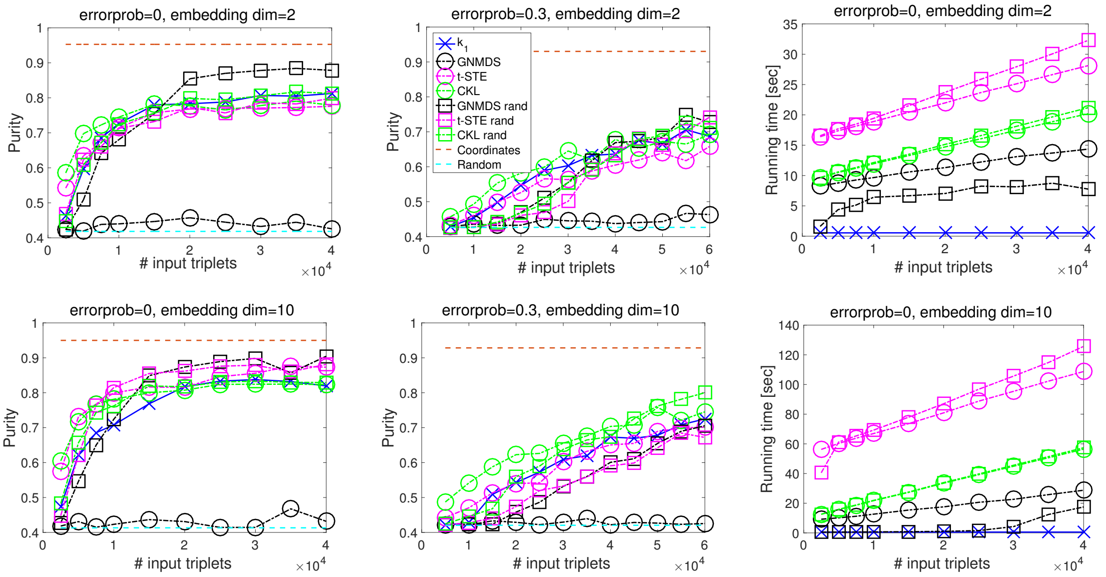


Figure 5.6: Landmark design for k_1 — 1000 points from USPS digits 1, 2, or 3 with Euclidean metric. First and second plot of a row: Purity as a function of the number of input similarity triplets for kernel K-means based on k_1 and for ordinary K-means applied to the ordinal embedding of one of the three embedding methods and to the original point set (brown line). The ordinal embedding algorithms were provided either the same input triplets as k_1 (curves with round markers) or a same number of randomly chosen triplets (square markers). The cyan line shows the purity of a random clustering as a lower baseline. Third plot: Corresponding running times in seconds for computing k_1 and the ordinal embeddings.

without any restriction, and incorrect with the same error probability *errorprob*.

Figure 5.6 shows the results in the landmark design for k_1 when clustering 1000 data points that were chosen uniformly at random from USPS digits 1, 2, or 3 (as in the experiments of Figure 5.5, but with a larger number of data points). The results for k_2 were slightly worse (plots omitted). From the 1000 data points we chose 15 landmark objects uniformly at random. We set the dimension of the space of the ordinal embeddings to two (1st row) or ten (2nd row). The first and the second plot of a row show the purity values of the various clusterings as a function of the number of input similarity triplets in case of $errorprob = 0$ and $errorprob = 0.3$, respectively. The third plot of a row shows the corresponding running times for computing k_1 and the ordinal embeddings in case of $errorprob = 0$. Based on the achieved purity values no method can be considered superior. The GNMDS algorithm apparently cannot deal properly with the landmark triplets. All methods require a higher number of input similarity triplets in case of $errorprob = 0.3$ than in case of $errorprob = 0$. Note that the ranges for the number of input triplets in Figure 5.6 are much smaller than those in Figure 5.5, although here the number of data points is more than twice as large as there. This shows that our kernel functions highly benefit from a landmark design. Just as it is the case for a general collection $\mathcal{S}$ of similarity triplets, our approach is highly superior regarding running time. For example, in case of $errorprob = 0$ and 20000 input triplets, the computation of k_1 took 0.5 seconds while the fastest embedding algorithm (GNMDS

Table 5.1: Characteristic values of All, All_reduced, T_All, and T_All_reduced.

	All	All_reduced	T_All	T_All_reduced
Number of similarity triplets	14194	12676	13514	12112
Number of unique similarity triplets	13152	12676	12502	12112
Number of unique similarity triplets in percent of all similarity triplets $[60 \cdot \binom{59}{2} = 102660]$	0.13	0.12	0.12	0.12
Number of contradicting pairs of similarity triplets (in the sets of unique similarity triplets)	242	0	198	0

with random input triplets) ran for 7 seconds when the embedding dimension equaled two. When the embedding dimension equaled ten, for random input triplets, a local optimum was found much faster in the optimization underlying GNMDS (running time of one second), but it ran for 18 seconds when provided with the landmark triplets. Even more striking, the CKL algorithm and the t-STE algorithm ran for 33 and more than 80 seconds, respectively, in case of *errorprob* = 0, 20000 input similarity triplets, and the embedding dimension equaling ten.

5.3.2 Crowdsourced similarity triplets

We applied our kernel functions to the car data set introduced in Section 4.5.2. For the car data set we do not have similarity triplets in the first place, but statements of the kind $(\star)$ (compare with Section 1.3 or Section 4.1), from which we can derive similarity triplets via (4.1). We derived a collection of similarity triplets from each of the four collections All, All_reduced, T_All, and T_All_reduced of statements, and in this chapter we use the names All, All_reduced, T_All, or T_All_reduced to refer to the corresponding collection of similarity triplets. The numbers of (unique) similarity triplets in each collection and of contradicting pairs of triplets are provided in Table 5.1.

We used our kernel functions to apply kernel PCA (Schölkopf et al., 1999) to the car data set. Figure 5.7 shows the projection of the data set onto the first two kernel principal components when working with the kernel function k_2 and the similarity triplets in T_All. We obtained a similar result when using k_1 or $k_3^{1,1}$ or one of the collections All, All_reduced, or T_All_reduced instead (figures omitted). The figure looks quite reasonable, with the cars obviously arranged in groups according to the subclasses of sports cars (top left), ordinary cars (middle right), and off-road/sport utility vehicles (bottom left). Also within the subclasses there is some reasonable structure. For example, the race-like sports cars are located near to each other and close to the Formula One car and the red cars at the top are strikingly close. All outliers according to our definition in Section 4.5.2 are located in the very left bottom part of the figure. The computation of k_1 or k_2 on the car data set based on the similarity triplets in any of the four collections took about 0.05 seconds, whereas the computation of the ordinal embedding shown in Figure 4.18 took 3.7 seconds.

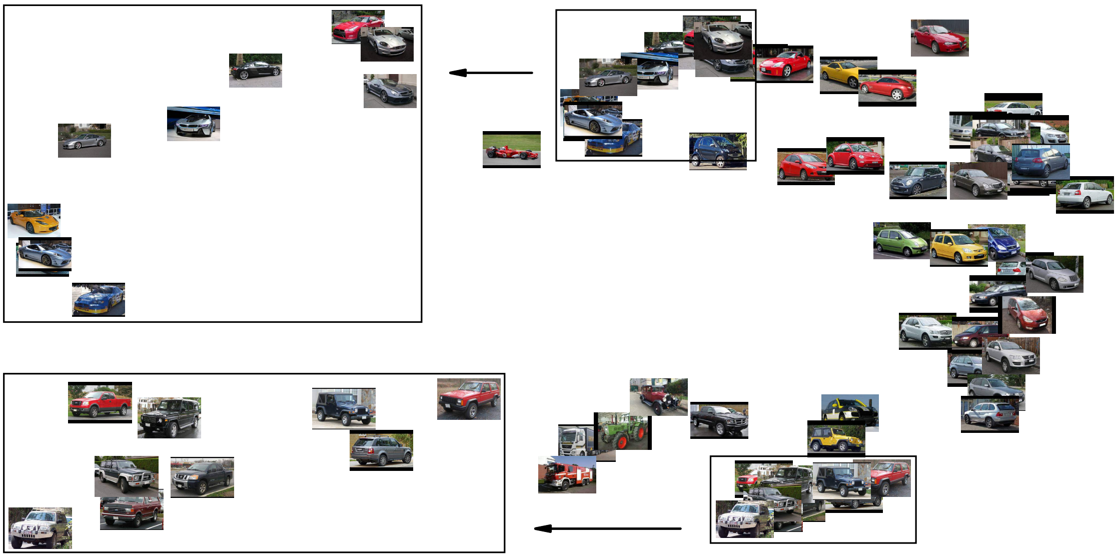


Figure 5.7: Kernel PCA on the car data set. Projection onto the first two principal components when using the kernel function k_2 based on the similarity triplets in T_All.

We also used our kernel functions to apply kernel K-means clustering to the car data set: after removing the four outliers, we wanted to recover the 56 remaining cars' grouping into ordinary cars, sports cars, and off-road/sport utility vehicles like we did in Section 4.5.2. Now one can proceed in two ways: One can either compute our kernel functions on the whole car data set, based on all similarity triplets in one of the collections All, All_reduced, T_All, or T_All_reduced, and restrict them to the 56 cars under consideration. Or one can compute our kernel functions only on the 56 cars, based only on similarity triplets comprising three of these cars and none of the removed outliers. For the sake of completeness we performed the experiment in both ways and present both results. The numbers of (unique) similarity triplets in each of the collections All, All_reduced, T_All, or T_All_reduced and of contradicting pairs of triplets after removing all triplets comprising an outlier as required for the second way are provided in Table 5.2. Table 5.3 shows the average purity (see equation (4.19) for its definition) of the various clusterings with respect to our assumed ground truth partitioning into ordinary cars, sports cars, and off-road/sport utility vehicles. The average is over 100 runs of the experiment. The clusterings obtained in different runs only differ due to the random initializations in kernel K-means. We set the number of replicates in kernel K-means to five and the maximum number of iterations to 100 like we did before, and we provided the correct number of clusters as input. Our first approach, that is computing the kernel function on the whole car data set and restricting it to the 56 cars under consideration, yielded marginally higher purity values than the other approach. The kernel function k_2 consistently achieved better results than the kernel function k_1. As expected, working with the similarity triplets in T_All or T_All_reduced led to higher purity values than working with the similarity triplets in All or All_reduced. Overall, the results are comparable to the ones that we obtained in Section 4.5.2 (compare with Table 4.4).

Table 5.2: Characteristic values of All, All_reduced, T_All, and T_All_reduced after removing the fire truck, the motortruck, the tractor, and the antique car from the car data set.

	All	All_reduced	T_All	T_All_reduced
Number of similarity triplets	11248	10242	10698	9772
Number of unique similarity triplets	10642	10242	10106	9772
Number of unique similarity triplets in percent of all similarity triplets $[56 \cdot \binom{55}{2} = 83160]$	0.13	0.12	0.12	0.12
Number of contradicting pairs of similarity triplets (in the sets of unique similarity triplets)	203	0	170	0

Table 5.3: Average purity (± standard deviation) of clusterings obtained from kernel K-means using k_1, k_2, or $k_3^{1,1}$ when clustering the 56 cars from the classes of ordinary cars, sports cars, and off-road/sport utility vehicles into three clusters (average over 100 runs). A kernel function k "restricted to 56 cars" (2nd to 4th row) means that we computed k on the whole car data set (comprising 60 cars), using all available similarity triplets in one of the collections All, All_reduced, T_All, or T_All_reduced, and restricted it to the 56 cars under consideration. A kernel function k "only on 56 cars" (5th to 7th row) means that we computed k only on the 56 cars, using only similarity triplets comprising three of these cars.

	All	All_reduced	T_All	T_All_reduced
k_1 restricted to 56 cars	0.77 (± 0.08)	0.76 (± 0.07)	0.80 (± 0.08)	0.80 (± 0.07)
k_2 restricted to 56 cars	0.90 (± 0.03)	0.89 (± 0.03)	0.91 (± 0.03)	0.89 (± 0.03)
$k_3^{1,1}$ restricted to 56 cars	0.87 (± 0.08)	0.86 (± 0.09)	0.91 (± 0.03)	0.90 (± 0.05)
k_1 only on 56 cars	0.77 (± 0.08)	0.74 (± 0.07)	0.81 (± 0.07)	0.79 (± 0.07)
k_2 only on 56 cars	0.89 (± 0.04)	0.88 (± 0.05)	0.89 (± 0.04)	0.88 (± 0.04)
$k_3^{1,1}$ only on 56 cars	0.86 (± 0.08)	0.85 (± 0.08)	0.89 (± 0.06)	0.87 (± 0.07)

5.4 Discussion

In this chapter we have proposed data-dependent kernel functions that can be evaluated when given only an arbitrary collection of similarity triplets for a data set $\mathcal{D}$. They can be used to apply any kernel method to $\mathcal{D}$ and hence provide a generic alternative to the ordinal embedding approach. Other than ordinal embedding algorithms our kernel functions are deterministic and we observed them to run significantly faster. Like for the algorithms that we proposed in Chapter 4, we believe that our kernel functions are applicable in situations in which ordinal embedding algorithms are not. A weakness of our kernel functions compared to an ordinal embedding approach is that, in general, they require a higher number of similarity triplets in order to produce a reasonable result. This comes from the fact that the value of our kernel functions at a pair of data points is affected only by appropriate pairs of similarity triplets. A means to increase the fraction of appropriate pairs in a collection $\mathcal{S}$ of similarity triplets, and thus to decrease the number of required input triplets, is a landmark design. In a landmark design, only dissimilarity comparisons (1.2) involving landmark objects are evaluated for creating $\mathcal{S}$. Our experiments showed that in a landmark design our kernel functions can compete with an ordinal embedding approach in terms of the required number of triplets.

Our work, which currently lacks a profound theoretical analysis, raises a number of questions: It would be nice to relate our kernel functions in some way to the underlying dissimilarity function ι, assuming our kernel functions are computed based on all similarity triplets, all of which are correct. For general data sets this will not be possible as our example presented in Section 5.2.5 shows. However, will this be possible if one makes strong assumptions on the data points (e.g., Euclidean data points sampled from a uniform distribution)? Another question is: how many similarity triplets are necessary, and how many incorrect ones are allowed, such that our kernel functions based on the given triplets are "close" to our kernel functions based on all similarity triplets, all of which are correct? In a landmark design, we would like to know what the optimal number of landmark objects is. How should we choose the landmark objects if additional knowledge about $\mathcal{D}$ is available? We suspect that the answers to the latter questions strongly depend on the task at hand. An obvious question concerns the implementation of our kernel functions. Currently, we have to explicitly compute the feature vectors $\Phi_{k_1}(x_i)$ or $\Phi_{k_2}(x_i)$, $i = 1, \ldots, n$, for computing the kernel matrix on $\mathcal{D} = \{x_1, \ldots, x_n\}$. As explained in Section 5.2.7, this leads to a high computational overhead that we would like to avoid. It is not clear to us whether it is really necessary to explicitly compute the feature vectors $\Phi_{k_1}(x_i)$ or $\Phi_{k_2}(x_i)$ or whether one can do better, for example, when assuming that the collection $\mathcal{S}$ of input similarity triplets comes in form of a matrix where each row corresponds to one triplet and the triplets are ordered in a specific way.

We hope that our work inspires future work: Our kernel functions are based on the idea of measuring similarity between two objects in $\mathcal{D}$ by comparing to which extent the two objects give rise to resembling similarity triplets. We came up with two ways of specifying what "resembling" similarity triplets are (corresponding to k_1 and k_2, respectively). However, there might be further possibilities, and this could result in another kernel functions based on similarity triplets. It is also natural to ask whether one can adapt our approach to other types of ordinal data (compare with Section 1.3). For example, how can we define meaningful kernel functions when given an arbitrary collection of answers to general dissimilarity comparisons of the form (1.1)?

Bibliography

S. Agarwal, J. Wills, L. Cayton, G. Lanckriet, D. Kriegman, and S. Belongie. Generalized non-metric multidimensional scaling. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2007.

C. Agostinelli and M. Romanazzi. Local depth of multivariate data. Technical report, Ca' Foscari University of Venice, 2008.

C. Agostinelli and M. Romanazzi. Local depth. *Journal of Statistical Planning and Inference*, 141(2):817–830, 2011.

P. Alestalo, D. A. Trotsenko, and J. Väisälä. Isometric approximation. *Israel Journal of Mathematics*, 125(1):61–82, 2001.

E. Amid and A. Ukkonen. Multiview triplet embedding: Learning attributes in multiple maps. In *International Conference on Machine Learning (ICML)*, 2015.

E. Amid, N. Vlassis, and M. Warmuth. *t*-exponential triplet embedding. arXiv:1611.09957 [cs.AI], 2016.

D. V. Andrade and L. H. de Figueiredo. Good approximations for the relative neighbourhood graph. In *Canadian Conference on Computational Geometry (CCCG)*, 2001.

D. Angluin and L. G. Valiant. Fast probabilistic algorithms for hamiltonian circuits and matchings. *Journal of Computer and System Sciences*, 18(2):155–193, 1979.

E. Arias-Castro. Some theory for ordinal embedding. arXiv:1501.02861 [math.ST], 2015.

Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, editors. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide.* Society for Industrial and Applied Mathematics, 2000.

V. Barnett and T. Lewis. *Outliers in Statistical Data.* Wiley, 1978.

R. Bartoszynski, D. K. Pearl, and J. Lawrence. A multidimensional goodness-of-fit test based on interpoint distances. *Journal of the American Statistical Association*, 92 (438):577–586, 1997.

F. S. Beckman and D. A. Quarles, Jr. On isometries of Euclidean spaces. *Proceedings of the American Mathematical Society*, 4(5):810–815, 1953.

R. Bennett. The intrinsic dimensionality of signal collections. *IEEE Transactions on Information Theory*, 15(5):517–525, 1969.

Y. Bilu and N. Linial. Monotone maps, sphericity and bounded second eigenvalue. *Journal of Combinatorial Theory, Series B*, 95(2):283–299, 2005.

M. Blum, R. W. Floyd, V. Pratt, R. L. Rivest, and R. E. Tarjan. Time bounds for selection. *Journal of Computer and System Sciences*, 7:448–461, 1973.

I. Borg and P. Groenen. *Modern Multidimensional Scaling: Theory and Applications.* Springer, 2005.

P. Bose, V. Dujmović, F. Hurtado, J. Iacono, S. Langerman, H. Meijer, V. Sacristán, M. Saumell, and D. R. Wood. Proximity graphs: E, δ, Δ, χ and ω. *International Journal of Computational Geometry and Applications*, 22(5):439–469, 2012.

F. C. Botelho, R. Pagh, and N. Ziviani. Simple and space-efficient minimal perfect hash functions. In *Workshop on Algorithms and Data Structures (WADS)*, 2007.

F. Camastra and A. Staiano. Intrinsic dimension estimation: Advances and open problems. *Information Sciences*, 328:26–41, 2016.

F. Camastra and A. Vinciarelli. Estimating the intrinsic dimension of data with a fractal-based method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24 (10):1404–1407, 2002.

I. Cascos. Data depth: Multivariate statistics and geometry. In W. S. Kendall and I. Molchanov, editors, *New Perspectives in Stochastic Geometry.* Oxford University Press, 2009.

C. Ceruti, S. Bassis, A. Rozza, G. Lombardi, E. Casiraghi, and P. Campadelli. DANCo: An intrinsic dimensionality estimator exploiting angle and norm concentration. *Pattern Recognition*, 47(8):2569–2581, 2014.

M. S. Chang, C. Y. Tang, and R. C. T. Lee. Solving the euclidean bottleneck matching problem by k-relative neighborhood graphs. *Algorithmica*, 8(1–6):177–194, 1992.

C. K. Chen and H. C. Andrews. Nonlinear intrinsic dimensionality computations. *IEEE Transactions on Computers*, C-23(2):178–184, 1974.

Y. Chen, X. Dang, H. Peng, and H. L. Bart, Jr. Outlier detection with the kernelized spatial depth function. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):288–305, 2009.

C. D. Correa and P. Lindstrom. Locally-scaled spectral clustering using empty region graphs. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2012.

J. A. Costa and A. O. Hero. Learning intrinsic dimension and intrinsic entropy of high-dimensional datasets. In *European Signal Processing Conference (EUSIPCO)*, 2004.

J. A. Costa, A. Girotra, and A. O. Hero. Estimating local intrinsic dimension with k-nearest neighbor graphs. In *IEEE Statistical Signal Processing Workshop (SSP)*, 2005.

N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and other Kernel-Based Learning Methods*. Cambridge University Press, 2000.

X. Dang and R. Serfling. Nonparametric depth-based multivariate outlier identifiers, and masking robustness properties. *Journal of Statistical Planning and Inference*, 140 (1):198–213, 2010.

J. Dattorro. *Convex Optimization and Euclidean Distance Geometry*. Meboo Publishing, 2005.

V. de Silva and J. B. Tenenbaum. Sparse multidimensional scaling using landmark points. Technical report, Stanford University, 2004.

I. Dhillon, Y. Guan, and B. Kulis. Kernel k-means, spectral clustering and normalized cuts. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2004.

R. T. Elmore, T. P. Hettmansperger, and F. Xuan. Spherical data depth and a multivariate median. In R. Y. Liu, R. Serfling, and D. L. Souvaine, editors, *Data Depth: Robust Multivariate Analysis, Computational Geometry and Applications*. American Mathematical Society, 2006.

B. Eriksson and M. Crovella. Estimating intrinsic dimension via clustering. In *IEEE Statistical Signal Processing Workshop (SSP)*, 2012.

A. Farahmand, C. Szepesvári, and J.-Y. Audibert. Manifold-adaptive dimension estimation. In *International Conference on Machine Learning (ICML)*, 2007.

L. C. Freeman. Centrality in social networks: Conceptual clarification. *Social Networks*, 1(3):215–239, 1978.

K. Fukunaga and D. R. Olsen. An algorithm for finding intrinsic dimensionality of data. *IEEE Transactions on Computers*, 20(2):176–183, 1971.

K. R. Gabriel and R. R. Sokal. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 18(3):259–278, 1969.

A. K. Ghosh and P. Chaudhuri. On maximum depth and related classifiers. *Scandinavian Journal of Statistics*, 32(2):327–350, 2005.

P. Grassberger and I. Procaccia. Measuring the strangeness of strange attractors. *Physica D: Nonlinear Phenomena*, 9(1–2):189–208, 1983.

D. Greene and P. Cunningham. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *International Conference on Machine Learning (ICML)*, 2006.

F. G. Gustavson. Two fast algorithms for sparse matrices: Multiplication and permuted transposition. *ACM Transactions on Mathematical Software*, 4(3):250–269, 1978.

T. Hagerup and T. Tholey. Efficient minimal perfect hashing in nearly minimal space. In *Symposium on Theoretical Aspects of Computer Science (STACS)*, 2001.

S. Haghiri, D. Ghoshdastidar, and U. von Luxburg. Comparison based nearest neighbor search. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

B. Hajek. *Random Processes for Engineers.* Cambridge University Press, 2015.

T. B. Hashimoto, Y. Sun, and T. S. Jaakkola. Metric recovery from directed unweighted graphs. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

H. Heikinheimo and A. Ukkonen. The crowd-median algorithm. In *Conference on Human Computation and Crowdsourcing (HCOMP)*, 2013.

E. Heim, M. Berger, L. M. Seversky, and M. Hauskrecht. Efficient online relative comparison kernel learning. In *SIAM International Conference on Data Mining (SDM)*, 2015.

M. Hein and J.-Y. Audibert. Intrinsic dimensionality estimation of submanifolds in Rˆd. In *International Conference on Machine Learning (ICML)*, 2005.

N. J. Higham. Exploiting fast matrix multiplication within the level 3 BLAS. *ACM Transactions on Mathematical Software*, 16(4):352–368, 1990.

T. Hofmann, B. Schölkopf, and A. J. Smola. Kernel methods in machine learning. *The Annals of Statistics*, 36(3):1171–1220, 2008.

L. Jain, K. G. Jamieson, and R. Nowak. Finite sample prediction and recovery bounds for ordinal embedding. In *Neural Information Processing Systems (NIPS)*, 2016.

K. G. Jamieson and R. Nowak. Low-dimensional embedding using adaptively selected ordinal data. In *Conference on Communication, Control, and Computing*, 2011.

J. W. Jaromczyk and G. T. Toussaint. Relative neighborhood graphs and their relatives. *Proceedings of the IEEE*, 80(9):1502–1517, 1992.

Y. Jiao and J.-P. Vert. The Kendall and Mallows kernels for permutations. In *International Conference on Machine Learning (ICML)*, 2015.

R. M. Johnson. Pairwise nonmetric multidimensional scaling. *Psychometrika*, 38(1): 11–18, 1973.

H. Kaplan, M. Sharir, and E. Verbin. Colored intersection searching via sparse rectangular matrix multiplication. In *Symposium on Computational Geometry (SoCG)*, 2006.

B. Kégl. Intrinsic dimension estimation using packing numbers. In *Neural Information Processing Systems (NIPS)*, 2002.

M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1–2):81–93, 1938.

M. Kleindessner and U. von Luxburg. Uniqueness of ordinal embedding. In *Conference on Learning Theory (COLT)*, 2014.

M. Kleindessner and U. von Luxburg. Dimensionality estimation without distances. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

M. Kleindessner and U. von Luxburg. Kernel functions based on triplet similarity comparisons. arXiv:1607.08456 [stat.ML], 2016.

M. Kleindessner and U. von Luxburg. Lens depth function and k-relative neighborhood graph: versatile tools for ordinal data analysis. *Journal of Machine Learning Research*, 18(58):1–52, 2017.

J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964a.

J. B. Kruskal. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29(2):115–129, 1964b.

J. Lawrence. *Interpoint Distance Methods for the Analysis of High Dimensional Data.* PhD thesis, The Ohio State University, 1996.

J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 2007. Data available on `https://snap.stanford.edu/data/`.

E. Levina and P. J. Bickel. Maximum likelihood estimation of intrinsic dimension. In *Neural Information Processing Systems (NIPS)*, 2004.

J. Li, J. A. Cuesta-Albertos, and R. Y. Liu. *DD*-classifier: Nonparametric classification procedure based on *DD*-plot. *Journal of the American Statistical Association*, 107 (498):737–753, 2012.

M. Li, X.-C. Lian, J. T.-Y. Kwok, and B.-L. Lu. Time and space efficient spectral clustering via column sampling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

S. Li. Concise formulas for the area and volume of a hyperspherical cap. *Asian Journal of Mathematics and Statistics*, 4(1):66–70, 2011.

C. Liu, K. Wu, and T. He. Sensor localization with ring overlapping based on comparison of received signal strength indicator. In *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS)*, 2004.

R. Y. Liu. On a notion of simplicial depth. *Proceedings of the National Academy of Sciences of the United States of America*, 85(6):1732–1734, 1988.

R. Y. Liu. On a notion of data depth based on random simplices. *The Annals of Statistics*, 18(1):405–414, 1990.

R. Y. Liu. Data depth and multivariate rank tests. In Y. Dodge, editor, *L1-Statistical Analysis and Related Methods*. North Holland, 1992.

R. Y. Liu, J. M. Parelius, and K. Singh. Multivariate analysis by data depth: descriptive statistics, graphics and inference. *The Annals of Statistics*, 27(3):783–840, 1999.

Z. Liu and R. Modarres. Lens data depth and median. *Journal of Nonparametric Statistics*, 23(4):1063–1074, 2011.

C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

M. J. McKemie and J. Väisälä. Spherical maps of Euclidean spaces. *Results in Mathematics*, 35(1):145–160, 1999.

K. Mosler. Depth statistics. In C. Becker, R. Fried, and S. Kuhnt, editors, *Robustness and Complex Data Structures: Festschrift in Honour of Ursula Gather*. Springer, 2013.

J. Opatrný. Total ordering problem. *SIAM Journal on Computing*, 8(1):111–114, 1979.

K. W. Pettis, T. A. Bailey, A. K. Jain, and R. C. Dubes. An intrinsic dimensionality estimator from near-neighbor information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(1):25–37, 1979.

S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

J. S. Sánchez, F. Pla, and F. J. Ferri. On the use of neighbourhood-based non-parametric classifiers. *Pattern Recognition Letters*, 18(11–13):1179–1186, 1997a.

J. S. Sánchez, F. Pla, and F. J. Ferri. Prototype selection for the nearest neighbour rule through proximity graphs. *Pattern Recognition Letters*, 18(6):507–513, 1997b.

I. J. Schoenberg. Metric spaces and completely monotone functions. *Annals of Mathematics*, 39(4):811–841, 1938.

B. Schölkopf, A. J. Smola, and K.-R. Müller. Kernel principal component analysis. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1999.

B. Schölkopf, J. Weston, E. Eskin, C. Leslie, and W. S. Noble. A kernel approach for learning from almost orthogonal patterns. In *European Conference on Machine Learning (ECML)*, 2002.

M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. In *Neural Information Processing Systems (NIPS)*, 2003.

R. Serfling. Depth functions in nonparametric multivariate inference. In R. Y. Liu, R. Serfling, and D. L. Souvaine, editors, *Data Depth: Robust Multivariate Analysis, Computational Geometry and Applications*. American Mathematical Society, 2006.

S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

B. Shaw and T. Jebara. Structure preserving embedding. In *International Conference on Machine Learning (ICML)*, 2009.

R. N. Shepard. The analysis of proximities: Multidimensional scaling with an unknown distance function. I. *Psychometrika*, 27(2):125–140, 1962a.

R. N. Shepard. The analysis of proximities: Multidimensional scaling with an unknown distance function. II. *Psychometrika*, 27(3):219–246, 1962b.

R. N. Shepard. Metric structures in ordinal data. *Journal of Mathematical Psychology*, 3(2):287–315, 1966.

R. N. Shepard and J. D. Carroll. Parametric representation of nonlinear data structures. In *International Symposium on Multivariate Analysis*, 1966.

C. R. Sherman. Nonmetric multidimensional scaling: A monte carlo study of the basic parameters. *Psychometrika*, 37(3):323–355, 1972.

J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

K. Sricharan, R. Raich, and A. O. Hero. Optimized intrinsic dimension estimator using nearest neighbor graphs. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010.

N. Stewart, G. D. A. Brown, and N. Chater. Absolute identification by relative judgment. *Psychological Review*, 112(4):881–911, 2005.

W. A. Sutherland. *Introduction to Metric and Topological Spaces*. Oxford Science Publications, 1975.

O. Tamuz, C. Liu, S. Belongie, O. Shamir, and A. T. Kalai. Adaptively learning the crowd kernel. In *International Conference on Machine Learning (ICML)*, 2011.

J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Y. Terada and U. von Luxburg. Local ordinal embedding. In *International Conference on Machine Learning (ICML)*, 2014. Code available on `https://cran.r-project.org/web/packages/loe`.

G. T. Toussaint. The relative neighbourhood graph of a finite planar set. *Pattern Recognition*, 12(4):261–268, 1980.

G. T. Toussaint. Applications of the relative neighbourhood graph. In *International Conference on Advances in Computing, Communication and Information Technology (CCIT)*, 2014.

G. T. Toussaint and C. Berzan. Proximity-graph instance-based learning, support vector machines, and high dimensionality: An empirical comparison. In *International Conference on Machine Learning and Data Mining (MLDM)*, 2012.

G. T. Toussaint, B. K. Bhattacharya, and R. S. Poulsen. The application of voronoi diagrams to non-parametric decision rules. In *Symposium on the Interface of Computing Science and Statistics*, 1984.

G. V. Trunk. Statistical estimation of the intrinsic dimensionality of data collections. *Information and Control*, 12(5):508–525, 1968.

J. W. Tukey. Mathematics and the picturing of data. In *International Congress of Mathematicians (ICM)*, 1974.

A. Ukkonen, B. Derakhshan, and H. Heikinheimo. Crowdsourced nonparametric density estimation using relative distances. In *Conference on Human Computation and Crowdsourcing (HCOMP)*, 2015.

G. Van Bever. *Contributions to Nonparametric and Semiparametric Inference based on Statistical Depth.* PhD thesis, Université libre de Bruxelles, 2013.

L. J. P. van der Maaten and K. Q. Weinberger. Stochastic triplet embedding. In *IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2012. Code available on `http://homepage.tudelft.nl/19j49/ste`.

A.W. van der Vaart. *Asymptotic Statistics.* Cambridge University Press, 1998.

V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer, 1998.

U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4): 395–416, 2007.

U. von Luxburg and M. Alamgir. Density estimation from unweighted k-nearest neighbor graphs: a roadmap. In *Neural Information Processing Systems (NIPS)*, 2013.

U. von Luxburg, A. Radl, and M. Hein. Hitting and commute times in large random neighborhood graphs. *Journal of Machine Learning Research*, 15:1751–1798, 2014.

M. J. Wilber, I. S. Kwak, and S. J. Belongie. Cost-effective hits for relative similarity comparisons. In *Conference on Human Computation and Crowdsourcing (HCOMP)*, 2014.

L. Xiao, R. Li, and J. Luo. Sensor localization based on nonmetric multidimensional scaling. In *International Conference on Sensing, Computing and Automation (ICSCA)*, 2006.

D. Yan, L. Huang, and M. I. Jordan. Fast approximate spectral clustering. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, 2009.

M. Yang. *Depth Functions, Multidimensional Medians and Tests of Uniformity on Proximity Graphs.* PhD thesis, The George Washington University, 2014.

F. W. Young. Nonmetric multidimensional scaling: Recovery of metric information. *Psychometrika*, 35(4):455–473, 1970.

F. W. Young. *Multidimensional Scaling: History, Theory, and Applications.* Lawrence Erlbaum Associates, 1987.

Y. Zuo and R. Serfling. General notions of statistical depth function. *The Annals of Statistics*, 28(2):461–482, 2000.