Advances in Data Analysis and Classification
https://doi.org/10.1007/s11634-025-00627-8

REGULAR ARTICLE

Weighted embedding and outlier detection of metric space data

Lauri Heinonen[1] · **Henri Nyberg**[1] · **Joni Virta**[1]

Received: 27 June 2024 / Revised: 20 December 2024 / Accepted: 9 February 2025

Abstract
This work discusses weighted kernel point projection (WKPP), a new method for embedding metric space or kernel data. WKPP is based on an iteratively weighted generalization of multidimensional scaling and kernel principal component analysis, and one of its main uses is outlier detection. After a detailed derivation of the method and its algorithm, we give theoretical guarantees regarding its convergence and outlier detection capabilities. Additionally, as one of our mathematical contributions, we give a novel characterization of kernelizability, connecting it also to the classical kernel literature. In our empirical examples, WKPP is benchmarked with respect to several competing outlier detection methods, using various different datasets. The obtained results show that WKPP is computationally fast, while simultaneously achieving performance comparable to state-of-the-art methods.

Keywords Distance matrix · Kernel method · Multidimensional scaling · Principal component analysis

Mathematics Subject Classification 62H25 · 62G35

1 Introduction

In modern data analysis, it is increasingly common to encounter data sets residing in a non-Euclidean space. To give just a few commonplace examples, in compositional

✉ Lauri Heinonen
lauri.k.heinonen@utu.fi

Henri Nyberg
henri.nyberg@utu.fi

Joni Virta
joni.virta@utu.fi

[1] Department of Mathematics and Statistics, University of Turku, 20014 Turku, Finland

Published online: 27 February 2025

data analysis (Pawlowsky-Glahn and Buccianti 2011) the observations are points in the unit simplex; in brain connectivity analysis (You and Park 2021) we observe a single positive-definite matrix per subject; in directional statistics (Mardia and Jupp 2000) the data live on the surface of a hypersphere.

When analyzing these and other non-Euclidean data, a key requirement is that the methods used should properly acknowledge the geometry of the ambient space, in order to guarantee reliable results. The methods in the literature of non-Euclidean data analysis can be divided roughly into two categories, depending on how they achieve this geometry-awareness. The first category contains methods that are highly specialized and targeted to one specific case of non-Euclidean data. For example, a large part of the methodologies in compositional data, directional statistics and functional data follow this idea. The second category contains general methods that depend on the sample only through the pair-wise distances of the points, the used distance function (metric) determining the geometry. As a highly non-exhaustive list of examples, see Bhattacharya and Patrangenaru (2003); Lyons (2013); Cornea et al. (2017); Dubey and Müller (2019); Petersen and Müller (2019); Virta et al. (2022).

In this work, we focus on the second category above and assume that our observed sample $x_1, \ldots, x_n$ resides in some metric space $(\mathcal{X}, d)$ (in the sequel we will use the term *metric space data* to refer to such datasets) and our methods are based on the corresponding (squared) distance matrix $\mathbf{D} = \{d^2(x_i, x_j)\}$. Despite the sometimes unusual geometric form of metric space data, many aspects of basic statistics transfer directly to their analysis. In particular, if the sample $x_1, \ldots, x_n$ contains outliers, these can skew the results of any non-robust statistical procedure, leading to false conclusions and non-generalizable findings. In some sense, this problem is even more acute for metric space data than for standard Euclidean data, since many metric datasets are too esoteric to even properly visualize, meaning that getting even a preliminary idea of whether a dataset *might* have outliers is already difficult.

Motivated by the previous, the primary purpose of this work is to develop outlier detection for metric space data. We achieve this by combining two statistical concepts: (i) multidimensional scaling (MDS) which is a classical method used for embedding metric space data to Euclidean spaces, see, e.g., Lee and Verleysen (2007), and (ii) k-step W-estimation, an iterative technique used to produce robust (i.e., outlier-resistant) estimators of location and scatter, see Hampel et al. (1986); Tyler (1987); Tyler et al. (2009). In essence, the proposed method, which we call *weighted kernel point projection* (WKPP), works by repeatedly embedding the observed metric space data into a Euclidean space, each time computing the weighted Mahalanobis distances of the data where the weighting is determined by the Mahalanobis distances from the previous iteration of the process. If a decreasing weight function is chosen, the effect of outliers gradually diminishes with each iteration, leading to a robust procedure. After a pre-specified number of iterations, we terminate the process and the final weights can be used (i) as an outlyingness measure to rank the observations in terms of their outlyingness, and (ii) to produce a weighted/robust embedding of the points into a Euclidean space. Both concepts have also been studied earlier in the literature and we next review and contrast these earlier works with WKPP.

Regarding outlier detection for metric space data, Cholaquidis et al. (2023); Geenens et al. (2023) propose the concept of metric lens depth for measuring the centrality of a point with respect to a given distribution in $(\mathcal{X}, d)$. As depth is essentially the inverse of outlyingess (Zuo and Serfling 2000), this allows for outlier detection. Blouvshtein and Cohen-Or (2018) detect outliers by identifying points that violate the triangle inequality in a contaminated distance matrix. A drawback of both previous methods is that they need to iterate over all subsets of the full sample of size three, making their computation slow for larger values of n. WKPP avoids this by being based on a sequence of partial eigendecompositions, being computable significantly faster. A similar complexity is held by the graph-based outlier detection algorithm of Amagata et al. (2021). However, their method is unnecessarily rigid, requiring the *a priori* definition of an outlier in terms of a maximal tolerable distance from its nearest neighbours. WKPP does not require this kind of information, but instead produces observation-specific weights, which then allow the quantitative ranking of the points in terms of their outlyingness, as is typical for statistical outlier detection methods. By choosing a proper cut-off value, this ranking then lets us flag a certain set of observations as outlying, see Sect. 5. Finally, in contrast to all three works discussed in this paragraph, we provide general conditions under which WKPP is guaranteed to perfectly separate the outliers from the bulk of the data.

Regarding robust embedding, Cayton and Dasgupta (2006) study an L_1-version of MDS whose computation is based on constrained semidefinite programming. As the exact solving of the problem is very difficult, Zhou et al. (2020) convexify the problem by penalizing it in a suitable way. Forero and Giannakis (2012) define a generative model where non-outliers are characterized by zero values of a certain parameter and use MDS coupled with LASSO-type penalty to obtain a robust embedding (detecting the outliers in the process). However, unlike our proposed WKPP, none of the previous three consider the embedding of test points (i.e., points which are not part of the original data used to train the model), which is a crucial part of any embedding method when it is used for actual inference, and not just for visualization.

The algebra behind WKPP is closely connected to the "kernel trick" used to apply statistical methods in high-dimensional feature spaces without actually computing the relevant feature vectors, see Shawe-Taylor and Cristianini (2004). Related to this connection, as one of our contributions we derive sufficient and necessary conditions characterizing all statistics (i.e., "methods") on which the kernel trick can be applied. Interestingly, despite the seemingly simple form of this result, we were unable to locate it in the earlier literature on kernel methods.

As a byproduct of the link to kernels, WKPP can also be turned into a weighted version of kernel principal component analysis (KPCA). This is because the space generated by a kernel is a metric space, and our method can thus be applied to distance matrices on this space. In the sequel, we use the phrase *kernel data* to refer to situations where we are working with a kernel matrix $\mathbf{K} = \{\kappa(x_i, x_j)\}$ corresponding to some kernel function κ and sample of n observations $x_1, \ldots, x_n$. While, to our best knowledge, iterative weighting has not been conducted earlier in the context of MDS and metric space data, two earlier works have used it to obtain robust KPCA: Huang et al. (2009) perform robust KPCA by downweighting observations that are orthogonal to the leading PCs. However, this strategy fails when the data contain outliers

with very large magnitudes as such points tend to themselves determine the directions of the first PCs, meaning that the largest outliers get maximal weights. Our solution avoids this effect by basing the weighting on Mahalanobis distances, downweighting the outliers independently of their contributions to the PCs (note that Huang et al. (2009) suggested this as an alternative approach but did not pursue it). Duan et al. (2011) robustify KPCA by iteratively downweighting each observation by its influence on the KPCA-solution, i.e., by how much leaving the observation out changes the variances and directions of the PCs. However, neither of Huang et al. (2009); Duan et al. (2011) consider the projection of test points to the feature space, a key part of the practical implementation of the methods. Finally, from the kernel viewpoint, our proposal can also be viewed as a computationally more efficient alternative to the outlier detection method of Schreurs et al. (2021) who use the Mahalanobis distances corresponding to the minimum covariance determinant (MCD) to quantify outlyingness in the feature space.

To summarize, the main contributions of this work are:

- We prove a characterizing result giving sufficient and necessary conditions for a statistic to be kernelizable, i.e., subjectible to the kernel trick.
- We derive a closed-form solution and an iterative algorithm for the proposed WKPP-procedure of repeatedly embedding and reweighting the observed metric space data, producing a weighted/robust embedding along with a measure of outlyingness for the data. Additionally, we derive the formula for projecting also out-of-sample points to the final embedding. As a byproduct we also obtain a weighted version of kernel PCA.
- We derive sufficient conditions under which the 1-step version of WKPP is able to perfectly identify a tight cluster of outliers and formulate a probabilistic model under which these conditions are asymptotically satisfied.
- We prove sufficient conditions for the convergence of the iteration weights and also investigate the matter with simulations.
- We compare WKPP with several competing methods using different real data sets.

The paper is organized as follows. In Sect. 2 we propose a characterization of all methods for which one can develop a metric space or kernel version. In Sect. 3 we introduce WKPP. We show the relevant results in Euclidean case and then present the algorithm in non-Euclidean case. Then we present theorems regarding outlier detection capability and convergence of the weights, illustrating both with simulations. In Sect. 4 we show two data examples of using WKPP. The first one benchmarks WKPP against two competing methods using three different outlier detection data sets and the second one uses WKPP to detect outliers in image feature data. Further discussion and open questions are given in Sect. 5. Appendix A contains further details on maximal invariants that we use to prove Theorem 1 in Sect. 2. The proofs of all technical results are presented in Appendix B.

2 Characterization of kernelization

In this section, we let $\mathbf{X} \in \mathbb{R}^{n\times p}$ be a data matrix whose rows correspond to n observed p-variate vectors. We also let $\mathbf{K} := \mathbf{X}\mathbf{X}^\top \in \mathbb{R}^{n\times n}$ denote the Euclidean kernel matrix. The history of kernel methods begins from Support vector machines (Boser et al. 1992) and Kernel principal component analysis (Schölkopf et al. 1998). The key idea behind developing kernel methods is that one shows a way to fit a linear method (separating hyperplane, PCA etc.) using only the dot products of observations. That is, showing that it is sufficient to know only $\mathbf{K}$ (instead of $\mathbf{X}$) to apply the method. It follows that one can fit these methods to observations in non-standard or high-dimensional spaces achieved usually through non-linear transformations of the original variables if one is able to calculate the inner product of such space. This inner product is then calculated through kernel functions without knowing the transformed variables. Moreover, given a kernel function $\kappa : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}$, if the kernel is positive semi-definite, it corresponds, by the Moore-Aronszajn theorem, to some Hilbert space $\mathcal{H}$ and its inner product, see (Aronszajn 1950)(Hofmann et al. 2008, section 2.2). Such spaces are called reproducing kernel Hilbert spaces (RKHS).

When speaking of kernel methods, we use the word *method* to mean the same as a *statistic* which is some quantity $t(\mathbf{X})$ calculated from the data $\mathbf{X}$. This means that, for example, by principal component analysis, we mean calculating principal component scores. Let next $\mathbf{H} := \mathbf{I}_n - (1/n)\mathbf{1}_n\mathbf{1}_n^\top \in \mathbb{R}^{n\times n}$ denote the centering matrix where $\mathbf{1}_n$ is the n-dimensional vector of ones. In the classical kernel methodology, the interest is on methods/statistics that can be *kernelized* in the sense of the following definition.

Definition 1 We say that a statistic $t(\mathbf{X})$ can be kernelized if there exists a mapping g such that $t(\mathbf{X}) = g(\mathbf{H}\mathbf{X}\mathbf{X}^\top\mathbf{H})$.

That is, a method can be kernelized if it can be written as a function of the double centered Euclidean kernel matrix $\mathbf{HKH}$ only. The centering via $\mathbf{H}$ has been included in our definition as almost all kernel methods use it, see the end of this section for a version without it. A method which can be kernelized then admits a kernel version which is obtained by replacing the Euclidean dot product matrix $\mathbf{K}$ with an arbitrary kernel matrix $\{\kappa(\mathbf{x}_i, \mathbf{x}_j)\}_{i,j=1}^n$.

Not all methods can be kernelized, e.g., the sample mean $t(\mathbf{X}) = \bar{\mathbf{x}}$ does not admit a kernelization as it is easy to find two pairs of points/vectors with the same angle between them (and therefore same value of the dot product), but which lie in different part of the space (and therefore have different means), leading to the data having identical matrices $\mathbf{HKH}$ but different means. In this section, we present a characterization of all methods that can be kernelized. This result seems classical, but we have not found it in any previous literature. The proofs are based on the theory of *maximal invariants* (Lehmann and Romano 2005). This theory is usually used to produce invariant tests but we give it here a different kind of use, and an additional short introduction to the theory of maximal invariants is included in Appendix A.

Theorem 1 *Let $t : \mathbb{R}^{n\times p} \to \mathcal{X}$ be a statistic taking values in some space $\mathcal{X}$. Then, $t(\boldsymbol{X})$ can be kernelized if and only if*

$$t(\boldsymbol{XV} + \boldsymbol{1}_n\boldsymbol{b}^\top) = t(\boldsymbol{X}),$$

for all $\boldsymbol{X} \in \mathbb{R}^{n\times p}$, *orthogonal* $\boldsymbol{V} \in \mathbb{R}^{p\times p}$ *and* $\boldsymbol{b} \in \mathbb{R}^p$.

Theorem 1 says that a method admits a kernel version if and only if it is unchanged under rotations, reflections and translations of the observed data. This means that, to establish kernelizability, one does not have to directly show that the method can be written using only dot products, but rather it is enough to show that it is invariant under certain transformations. As an example, consider ridge regression, which computes the predictions $\hat{\mathbf{y}} = \mathbf{H}\mathbf{X}\hat{\boldsymbol{\beta}}^R = \mathbf{H}\mathbf{X}(\mathbf{X}^\top\mathbf{H}\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{H}\mathbf{X}^\top\mathbf{y}$. The desired invariance can be seen with direct substitution, see Appendix A, implying the well-known fact that ridge regression can be kernelized. As another example, the kernelized MRCD outlier detection method developed in Schreurs et al. (2021) could be verified to satisfy the condition in Theorem 1. Similarly, the computation of PCA scores based on any scatter matrix can be kernelized, and this will be considered in detail in the next section.

In the classical kernel methodology literature, kernelizability is often approached through the concept of dual problems, see Boser et al. (1992); Cristianini and Shawe-Taylor (2004). This relates to scenarios where we are interested in linear combinations $z_i = \mathbf{w}(\mathbf{X})'(\mathbf{x}_i - \bar{\mathbf{x}})$, where the weight vector $\mathbf{w}(\mathbf{X}) \in \mathbb{R}^p$ is estimated from the training data. Classical kernel methods, such as kernel SVM or kernel ridge regression, take precisely this form and are based on the observation that, instead of first computing $\mathbf{w}(\mathbf{X})$ and then projecting the data onto it (primal problem), it is possible to compute the projections z_i directly in a manner that uses the training data only through their pairwise inner products (dual problem). In this terminology, Theorem 1 thus states that the dual formulation exists precisely when the z_i are invariant to transformations $\mathbf{X} \mapsto \mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top$ of the data, and for this to hold, the weight vector must satisfy $\mathbf{w}(\mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top) = \mathbf{V}'\mathbf{w}(\mathbf{X})$ (it is straightforward to check that the weight functions for both ridge and SVM indeed satisfy this). However, Theorem 1 is not limited to quantities of the form $\mathbf{w}(\mathbf{X})'(\mathbf{x}_i - \bar{\mathbf{x}})$ but rather applies to any possible statistics computable from the data.

It is well-known that kernel methodology is less "interpretable" than the related linear methods in the sense that the former do not admit loadings/weights/coefficients that allow evaluating the contribution of each original variable to the obtained solution, see, e.g, Schölkopf et al. (1998). From an intuitive viewpoint, this is because the linear combinations are taken in the feature space, lacking explicit connection to the original variables. Theorem 1 now provides an alternative mathematical explanation to this phenomenon by stating that, e.g., PCA loadings are not kernelizable since they lack the invariance property required by Theorem 1. This idea is implicitly used in kernel literature when moving from the primal problem (involving the weights) to the dual problem (bypassing the weights and using directly the linear combinations).

Finally, inspection of the proof of Theorem 1 reveals that if we instead define kernelization based on the non-centered kernel $\mathbf{K} = \mathbf{X}\mathbf{X}^\top$, then the result of Theorem 1 continues to hold if $\mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top$ is replaced by $\mathbf{X}\mathbf{V}$.

3 Weighted kernel point projection

As is standard with kernel methodology, we first introduce the relevant theory in Euclidean case and derive the relevant formulas to fit our method using the dot products of the observations (Sect. 3.1). Then we proceed to the non-Euclidean case (that is, metric space data or kernel data) using the kernel trick to generalize the Euclidean version of the method to obtain the WKPP-algorithm (Sect. 3.2).

3.1 Euclidean formulation

Our aim is to develop a robust version of (kernel) principal component analysis/multidimensional scaling by weighting the observations. For this, we use the principal component analysis based on k-step W-estimates. For convenience, we assume that in all encountered eigendecompositions and singular value decompositions, the relevant eigenvalues and singular values are distinct. As stated above, we work throughout Sect. 3.1 under the assumption of a Euclidean data matrix $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)^\top \in \mathbb{R}^{n\times p}$.

Let $g : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a function. Below, we also use the notation $g(\mathbf{v})$ for a vector $\mathbf{v} \in \mathbb{R}^n_{\geq 0}$ to mean that the function g is taken componentwise from $\mathbf{v}$. Let us then define the iterated mean vector

$$\bar{\mathbf{x}}_{k+1} = \sum_{i=1}^{n} \frac{g_{k,i}}{\sum_j g_{k,j}} \mathbf{x}_i$$

and the iterated scatter matrix

$$\mathbf{S}_{k+1} = \sum_{i=1}^{n} \frac{g_{k,i}}{\sum_j g_{k,j}} (\mathbf{x}_i - \bar{\mathbf{x}}_{k+1})(\mathbf{x}_i - \bar{\mathbf{x}}_{k+1})^\top,$$

where $g_{k,i} = g\{(\mathbf{x}_i - \bar{\mathbf{x}}_k)^\top \mathbf{S}_k^{-1} (\mathbf{x}_i - \bar{\mathbf{x}}_k)\}$. By repeatedly computing these quantities, we obtain a sequence of mean and scatter estimates $(\bar{\mathbf{x}}_1, \mathbf{S}_1)$, $(\bar{\mathbf{x}}_2, \mathbf{S}_2)$, ... In the absence of any auxiliary information, a reasonable starting point for the iteration is the case of equal weights, $g_{0,i} \equiv 1$. Convergence of the weights, regardless of starting point, is proved for one particular case in Sect. 3.4. Under the choice $g_{0,i} \equiv 1$, the initial estimates $\bar{\mathbf{x}}_1$ and $\mathbf{S}_1$ are the regular mean vector and covariance matrix. For future purposes, we define

$$\mathbf{w}_k = \frac{1}{\sum_i g_{k,i}} (g_{k,1}, \ldots, g_{k,n}).$$

The function g is used to weight the observations based on their Mahalanobis distance from the mean. Increased robustness is achieved by choosing g to be decreasing, thus downweighting the outliers. This is typical in the robust literature (Hampel et al. 1986), and leads to the outliers being recognizable as the observations with the smallest weights. The role of the iteration is to help better recognize the outliers by gradually

decreasing their misleading effect on the mean and the covariance matrix, and thus on the Mahalanobis distance. We also allow g to be increasing, in which case outliers are given larger weights. This strategy is often used in projection pursuit to detect outliers (Reza and Ruhi 2015; Fischer et al. 2021). In this article, decreasing g is therefore proposed for robust embedding and increasing g for outlier detection, although one could be able to find interesting uses for other choices. For example, using increasing weights for embedding would intuitively lead to an embedding space that represents extreme outliers well (and possibly represents the bulk badly), which could be useful if one wants to, e.g., search for subgroups of outliers.

All matrices $\mathbf{S}_k$ produced by the above iteration are *scatter matrices*. A scatter matrix is any matrix $\mathbf{S}(\mathbf{X}) \in \mathbb{R}^{p\times p}$ calculated from a data matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ so that for any matrix $\mathbf{C} \in \mathbb{R}^{p\times p}$ and any vector $\mathbf{b} \in \mathbb{R}^p$ we have $\mathbf{S}(\mathbf{X}\mathbf{C}^\top + \mathbf{1}_n\mathbf{b}^\top) = \mathbf{C}\mathbf{S}(\mathbf{X})\mathbf{C}^\top$. The basic example is the regular covariance matrix, which is obtained with the choice $g(x) \equiv 1$ above, and other scatter matrices are obtained with different choices of g. Some typical contexts to encounter scatter matrices are in invariant coordinate selection (ICS) (Tyler et al. 2009) or as alternatives to covariance matrix in various plug-in procedures (Nordhausen and Tyler 2015).

The principal component analysis based on matrix $\mathbf{S}_k$ is done by calculating its eigendecomposition $\mathbf{S}_k = \mathbf{E}\mathbf{\Lambda}\mathbf{E}^\top$ and then calculating the principal component score matrix $\mathbf{Z} = (\mathbf{X} - \mathbf{1}_n\bar{\mathbf{x}}_k^\top)\mathbf{E}$. An embedding to lower dimensional (say q-dimensional) space can be achieved by choosing only the q first columns of the matrix $\mathbf{Z}$ as our new coordinates. The primary objective of this section is to kernelize this procedure, i.e., PCA based on the iteratively weighted scatter matrix $\mathbf{S}_k$, in a manner described in Sect. 2, and the next theorem shows that this can indeed be done. This theorem links this Sect. 3 to the general analysis of Sect. 2 by giving one concrete example of applying Theorem 1. The value of Theorem 2 lies in the fact that for several scatter matrices $\mathbf{S}(\mathbf{X})$, e.g., Tyler's M-estimator (Tyler 1987), verifying the required equivariance property is a much simpler task than directly trying to express the resulting principal component scores via the dot products of the data.

Theorem 2 *Let $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ be a centered data matrix and let $\boldsymbol{S}(\boldsymbol{X}) \in \mathbb{R}^{p\times p}$ be a scatter matrix. The principal component scores produced by PCA based on $\boldsymbol{S}(\boldsymbol{X})$ are invariant under orthogonal transformations and translations and therefore kernelizable.*

Theorem 2 does not tell us how to obtain the kernelization (only that it can be done) and, hence, we next derive it. We do this by expressing the principal components $\mathbf{Z}$ as a function of the matrix $\mathbf{A}$ defined as

$$\mathbf{A} = -\frac{1}{2}\mathbf{H}\mathbf{D}\mathbf{H} = \mathbf{H}\mathbf{K}\mathbf{H} \in \mathbb{R}^{n\times n}, \tag{1}$$

where $\mathbf{D} := \{\|\mathbf{x}_i - \mathbf{x}_j\|^2\}_{i,j=1}^n$ is the matrix of pairwise squared Euclidean distances and $\mathbf{K} = \{\mathbf{x}_i^\top\mathbf{x}_j\}_{i,j=1}^n$ is the matrix of Euclidean inner products. The above connection between $\mathbf{D}$ and $\mathbf{K}$ (which is well-known, see (Mardia et al. 1979, 14.2.1)) is convenient as it allows us to later generalize the method to both metric space data and kernel data using a single set of formulas.

Let next the centered data matrix have the truncated singular value decomposition $\mathbf{HX} = \mathbf{U\Pi V}^\top$, where $\mathbf{U} \in \mathbb{R}^{n\times p}$ has orthonormal columns, $\mathbf{\Pi} \in \mathbb{R}^{p\times p}$ is diagonal and $\mathbf{V} \in \mathbb{R}^{p\times p}$ is an orthogonal matrix. The main idea behind the following proofs is to show that the principal components $\mathbf{Z}$ can be expressed as a function of the matrices $\mathbf{U}$ and $\mathbf{\Pi}$. And, even though we do not observe $\mathbf{HX}$, the matrices $\mathbf{U}$ and $\mathbf{\Pi}$ can still be obtained from the eigendecomposition $\mathbf{U\Pi}^2\mathbf{U}^\top$ of the matrix $\mathbf{A} = \mathbf{HXX}'\mathbf{H}$, thus letting us compute $\mathbf{Z}$ solely based on $\mathbf{A}$.

Another matrix that appears frequently in the following results is $\mathbf{H}_k = \mathbf{I}_n - \mathbf{w}_k\mathbf{1}_n^\top \in \mathbb{R}^{n\times n}$, where $\mathbf{w}_k$ denotes the weight vector of the kth iteration. It is simple to check that $\mathbf{H}_k$ is idempotent but not symmetric, and therefore an oblique projection matrix. The next lemma shows that $\mathbf{H}_k$ can be used to center observations by their $\mathbf{w}_k$-weighted mean.

Lemma 1 *Given a vector $\boldsymbol{w}_k \in \mathbb{R}^n$ of weights summing to one, let $\boldsymbol{H}_k = \boldsymbol{I}_n - \boldsymbol{w}_k\boldsymbol{I}_n^\top$. Then, the* i*th row of the matrix $\boldsymbol{H}_k^\top\boldsymbol{X}$ is*

$$\boldsymbol{x}_i - \sum_{j=1}^{n} w_{k,j}\boldsymbol{x}_j.$$

We divide the treatment of the full procedure into two parts: first, Theorem 3 provides the desired expression for the iterative updating of the weight vector $\mathbf{w}_k \in \mathbb{R}^n$ and, afterwards, in Theorem 4 we consider the projection of a test point $\mathbf{x} \in \mathbb{R}^p$ given the already computed weights.

Theorem 3 *Fix* k *and let $\boldsymbol{w}_k \in \mathbb{R}^n$ denote the weight vector of the* k*th iteration. Then,*

$$w_{k+1,i} = \frac{g(b_i)}{\sum_{j=1}^n g(b_j)},$$

where

$$b_i = \left(\boldsymbol{H}_k^\top\boldsymbol{U}\Big[\boldsymbol{U}^\top\{diag(\boldsymbol{w}_k) - \boldsymbol{w}_k\boldsymbol{w}_k^\top\}\boldsymbol{U}\Big]^{-1}\boldsymbol{U}^\top\boldsymbol{H}_k\right)_{ii}.$$

To obtain the weight vector $\mathbf{w}_k \in \mathbb{R}^n$ for an arbitrary iteration k, one starts from $\mathbf{w}_0 := \frac{1}{n}\mathbf{1}_n$ and repeatedly applies the update formula in Theorem 3. The procedure is formulated as an algorithm later in Sect. 3.2 where we discuss WKPP, the extension of the method to non-Euclidean scenarios.

Let now $\mathbf{w} \in \mathbb{R}^n$ denote a fixed weight vector (computed with the iterative procedure). Next, we show how to compute the centered principal component scores for an arbitrary test point $\mathbf{x} \in \mathbb{R}^p$ in PCA based on $\mathbf{w}$-weighted scatter matrix. For this, we let $d(\mathbf{x}) = \{\|\mathbf{x} - \mathbf{x}_i\|^2\}_{i=1}^n \in \mathbb{R}^n$ be the vector of pairwise squared Euclidean distances between $\mathbf{x}$ and the training sample $\mathbf{x}_1, \ldots, \mathbf{x}_n$. In the case of metric space data, $d(\mathbf{x})$ contains precisely the information that is available about a test point $\mathbf{x}$. Similarly, let $k(\mathbf{x}) = \{\mathbf{x}^\top\mathbf{x}_i\}_{i=1}^n \in \mathbb{R}^n$ gather the pairwise inner products between $\mathbf{x}$ and the training

points, corresponding to the full information one has about a test point when working in a kernel data scenario.

We further define the "joint" quantity (cf. $\mathbf{A}$ in formula (1)),

$$a(\mathbf{x}) := -\frac{1}{2}\mathbf{H}\left\{d(\mathbf{x}) - \frac{1}{n}\mathbf{D}\mathbf{1}_n\right\} = \mathbf{H}\left\{k(\mathbf{x}) - \frac{1}{n}\mathbf{K}\mathbf{1}_n\right\}, \tag{2}$$

where the choice of the formula again depends on whether one works with distances or with inner products (and as in (1) the two formulas are equivalent).

Theorem 4 *Let $\boldsymbol{w} = (w_1, \ldots, w_n)^\top \in \mathbb{R}^n$ contain non-negative weights summing to one. Then the centered principal component scores of $\boldsymbol{x} \in \mathbb{R}^p$ in PCA based on $\boldsymbol{w}$-weighted scatter matrix are*

$$z(\boldsymbol{x}) = \boldsymbol{M}^\top \boldsymbol{\Pi}^{-1}\boldsymbol{U}^\top a(\boldsymbol{x}) - \boldsymbol{M}^\top \boldsymbol{\Pi}\boldsymbol{U}^\top \boldsymbol{w}, \tag{3}$$

where $\boldsymbol{M} \in \mathbb{R}^{p\times p}$ has the eigenvectors of the matrix $\boldsymbol{\Pi}\boldsymbol{U}^\top\{\boldsymbol{H}_k diag(\boldsymbol{w})\boldsymbol{H}_k^\top\}\boldsymbol{U}\boldsymbol{\Pi}$ as its columns.

In most practical scenarios one is interested in the projections of the training points, and therefore we next give a separate, simplified formula for them.

Corollary 1 *The centered principal component scores of the training data $\boldsymbol{X}$ in PCA based on $\boldsymbol{w}$-weighted scatter matrix are*

$$\boldsymbol{Z} = \boldsymbol{H}_k^\top \boldsymbol{U}\boldsymbol{\Pi}\boldsymbol{M} \in \mathbb{R}^{n\times p},$$

where $\boldsymbol{M}$ is as in Theorem 4.

In the next section, we generalize the obtained weighted PCA procedure to the case of non-Euclidean data. We achieve this by applying the results above to a matrix $\mathbf{A}$ obtained from non-Euclidean data. Such "plug-in" procedures have long been successfully used, starting from the classical multidimensional scaling and including several more recent proposals such as Lyons (2013); Dubey and Müller (2019); Dai and Lopez-Pintado (2023). Note also that, since the plugging-in itself is heuristic, the usefulness of the resulting method needs to be verified separately. We accomplish this in the following both by deriving theoretical guarantees for the method (Theorems 5–7) under the assumption of arbitrary non-Euclidean $\mathbf{A}$ and via numerous simulation and real data examples.

3.2 Non-Euclidean formulation

Next we present our proposed WKPP-method for the weighted embedding of metric space/kernel data. For this, we assume that one has observed either (a) the matrix $\mathbf{D} = \{d^2(x_i, x_j)\}_{i,j=1}^n$ of pairwise squared distances between some set of n objects $x_1, \ldots, x_n$, or (b) the matrix $\mathbf{K} = \{\kappa(x_i, x_j)\}_{i,j=1}^n$ of kernel-based inner products of

some set of n objects. Note that, in both cases, the objects $x_1, \ldots, x_n$ do not need to be Euclidean. Depending on the case, let $\mathbf{A}$ denote either the matrix $-(1/2)\mathbf{HDH}$ or $\mathbf{HKH}$. Similarly, given a test point $\mathbf{x}$, let $a(\mathbf{x})$ be defined using either of the formulas in (2).

We use the notation of the previous section and present WKPP in Algorithm 1 as a direct generalization of those results. The matrices $\mathbf{U} \in \mathbb{R}^{n\times p}$ and $\mathbf{\Pi} \in \mathbb{R}^{p\times p}$ in the algorithm are taken from the truncated eigendecomposition $\mathbf{A} = \mathbf{U}\mathbf{\Pi}^2\mathbf{U}^\top$ of the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$. This truncation implicitly assumes that all relevant information is contained in the first $p \leq n$ eigenvectors and eigenvalues, where p is chosen apriori, see Sect. 5 for discussion on how to choose it. This assumption holds exactly in the Euclidean case when p is chosen equal to the number of variables.

When the constant function $g(x) = 1$ is used, Algorithm 1 reduces to multidimensional scaling or kernel PCA, depending on whether metric space data or kernel data is used. The vector $\mathbf{b}$ in Algorithm 1 contains the squared Mahalanobis distances between the weighted p-dimensional embeddings of the observations and their mean embedding. As such, the full procedure can be seen as repeatedly embedding the data, computing a measure of outlyingness (i.e., the weights) from the embedding and then re-weighting the observations based on their outlyingness.

Algorithm 1 Weighted kernel point projection

Data: matrix $\mathbf{A}$ of the training data
Parameters: weight function g
dimension parameter p
number of iterations N
Input: vector $a(\mathbf{x})$ of a test point
Result: projected training point $z(\mathbf{x})$
$\mathbf{U}$ is the matrix of p first eigenvectors of $\mathbf{A}$
$\mathbf{w} = \frac{1}{n}\mathbf{1}_n$ (initial value).
$\mathbf{H} = \mathbf{I}_n - \mathbf{w}\mathbf{1}_n^\top$
for $i \in \{1, \ldots, N\}$ **do**
  $\mathbf{b} = \mathrm{diag}(\mathbf{H}^\top\mathbf{U}[\mathbf{U}^\top\{\mathrm{diag}(\mathbf{w}) - \mathbf{w}\mathbf{w}^\top\}\mathbf{U}]^{-1}\mathbf{U}^\top\mathbf{H})$
  $\mathbf{w} = (g(b_1), \ldots, g(b_n))/\sum_j g(b_j)$
  $\mathbf{H} = \mathbf{I}_n - \mathbf{w}\mathbf{1}_n^\top$
$\mathbf{\Pi}$ is a diagonal matrix of p first eigenvalues of $\mathbf{A}$
$\mathbf{M}$ is the matrix of p first eigenvectors of $\mathbf{\Pi}\mathbf{U}^\top\mathrm{diag}(\mathbf{w})\mathbf{U}\mathbf{\Pi}$
return $z(\boldsymbol{x}) = \boldsymbol{M}^\top\mathbf{\Pi}^{-1}\boldsymbol{U}^\top a(\boldsymbol{x}) - \boldsymbol{M}^\top\mathbf{\Pi}\boldsymbol{U}^\top\boldsymbol{w}$

3.3 Outlier detection guarantee

As described in Sects. 3.1 and 3.2, the direct purpose of WKPP is the weighted embedding of points into a Euclidean space. However, as a byproduct of this process, it also performs outlier detection. Namely, by taking a weight function g that is increasing (decreasing), we expect that any outliers in the sample are given large (small) weights. Depending on the used weight function, outliers can thus be detected as those indices

which have the largest/smallest values in the final weight vector $\mathbf{w} \in \mathbb{R}^n$, see Sect. 5 for some strategies on choosing an appropriate cut-off value.

We next give sufficient conditions under which the previous process is able to perfectly separate the outliers from the bulk, in case of any strictly increasing weight function. While we consider, for simplicity, a case with one cluster of outliers, similar techniques could be used to derive similar results for a larger number of clusters, with the cost of more cluttered assumptions and proof. To present this result, we first define some notation. Let the non-zero matrix $\mathbf{D} \in \mathbb{R}^{n \times n}$ of pairwise squared distances in an arbitrary metric space be divided into blocks as

$$\begin{pmatrix} \mathbf{D}_{11} & \mathbf{D}_{12} \\ \mathbf{D}_{12}^\top & \mathbf{D}_{22} \end{pmatrix},$$

where the diagonal blocks are square and have the sizes $n_1 \times n_1$ and $n_2 \times n_2$, respectively. The underlying idea is that the group of size n_1 represents the "bulk" (non-outliers) and that there are a total of n_2 outliers. Denote by $\bar{d}_{11} := \mathbf{1}_{n_1}^\top \mathbf{D}_{11} \mathbf{1}_{n_1} / n_1^2$, $\bar{d}_{12} := \mathbf{1}_{n_1}^\top \mathbf{D}_{12} \mathbf{1}_{n_2} / (n_1 n_2)$, $\bar{d}_{22} := \mathbf{1}_{n_2}^\top \mathbf{D}_{22} \mathbf{1}_{n_2} / n_2^2$ the means of the three blocks and let $\tilde{\mathbf{D}}$ be the block matrix

$$\tilde{\mathbf{D}} := \begin{pmatrix} \bar{d}_{11} \mathbf{J}_{n_1, n_1} & \bar{d}_{12} \mathbf{J}_{n_1, n_2} \\ \bar{d}_{12} \mathbf{J}_{n_2, n_1} & \bar{d}_{22} \mathbf{J}_{n_2, n_2} \end{pmatrix},$$

where $\mathbf{J}_{n_1, n_2}$ denotes a matrix of size $n_1 \times n_2$ full of ones. Note that, since outlier detection is an unsupervised task, the scenario does not admit the "training/test" division with labeled training data, meaning also that the problem cannot be approached with supervised classification methodology such as linear or quadratic discriminant analysis.

Theorem 5 below makes the implicit assumption that the distance matrix $\mathbf{D}$ of the data can be approximated by the matrix $\tilde{\mathbf{D}}$ in a *relative* sense. That is, the between-group variation between the bulk and the outliers must be large enough compared to the corresponding within-group variations, which is a natural requirement for the successful detection of outliers.

We denote by $w_1, \ldots, w_n \in [0, 1]$ the weights calculated with WKPP in Algorithm 1 using $\mathbf{A} = -(1/2)\mathbf{HDH}$ as the input, with the dimension parameter $p = 1$, number of iterations $N = 1$ and any strictly increasing weight function g. The restriction to a single iteration is done for simplicity, making the estimate a so-called "one-step W-estimate" in classical terms. Thus, a perfect separation of outliers and the bulk is obtained if $\max\{w_1, \ldots, w_{n_1}\} < \min\{w_{n_1+1}, \ldots, w_n\}$. We next give sufficient conditions for this to hold. Note that the result is both non-probabilistic and non-asymptotic.

Theorem 5 *Let $\mathbf{D}$, $\tilde{\mathbf{D}}$ and $w_1, \ldots, w_n$ be as described above. Assume that $\bar{d}_{11}, \bar{d}_{12}, \bar{d}_{22} > 0$ and that $\bar{d}_{12} > \max\{\bar{d}_{11}, \bar{d}_{22}\}$. Assume further that $n_1 > n_2$ and*

that

$$\frac{\|\boldsymbol{D}-\tilde{\boldsymbol{D}}\|_{op}}{(\bar{d}_{12}-\frac{\bar{d}_{11}+\bar{d}_{22}}{2})} \leq \frac{1}{8^{1/2}n^{3/2}} n_1^{1/2} n_2^{1/2}(n_1-n_2). \tag{4}$$

where the operator norm $\|\cdot\|_{op}$ *is the largest singular value.*

Then,

$$\max\{w_1, \ldots, w_{n_1}\} < \min\{w_{n_1+1}, \ldots, w_n\}.$$

By looking at Theorem 5, we see that a larger distance between outliers and the bulk makes the outliers easier to detect. The denominator on the left-hand-side represents the difference of between-group and within-group distances. This is somewhat intuitive, so in the following simulation we fix the distance to a single large enough value. By setting the proportion of outliers to be ε (and assuming all else stays unchanged), which gives $n_1 = (1-\varepsilon)n$ and $n_2 = \varepsilon n$, we get the right-hand side of (4) to be $\sqrt{n/8}\sqrt{\varepsilon(1-\varepsilon)}(1-2\varepsilon)$, which is maximized when $\varepsilon \approx 0.1465$. This indicates that the most efficient outlier detection is achieved with not too many but not too few outliers. Theorem 5 has quite strict assumptions, but the result, about the perfect separation of the bulk and outliers, is also strong. As the quantities involved are continuous in the data, one could speculate that when the assumptions are almost met, the separation result can be almost achieved. In the robust statistics literature, outliers are often modeled as point mass contaminations (Hampel et al. 1986) that are located far away from the bulk. Hence, they represent data that come from a completely different population and have entered the observed data by accident, and the setting of Theorem 5 thus mimics this idealized situation.

To demonstrate Theorem 5, we ran a simulation where we generated a dataset with n observations with p variables. Proportion $1-\varepsilon$ (bulk) of the observations come from $\mathcal{N}_p(\mathbf{0}, \mathbf{I}_p)$ and proportion ε (outliers) come from $\mathcal{N}_p(10 \cdot \mathbf{1}_p, \mathbf{I}_p)$. We had two cases for the relation of n and p: fixed p with $(n,\ p) = (50,\ 40)$, $(60,\ 40)$, $(70,\ 40)$ and relation $p = 0.8n$ with $(n,\ p) = (50,\ 40)$, $(100,\ 80)$, $(200,\ 160)$, $(400,\ 320)$ (notice that $n = 50$, $p = 40$ belongs to both cases). Euclidean distance was used and in Fig. 1 we have plotted proportions of the cases where the inequality (4) holds (for 5000 repetitions) as a function of ε for all considered values of (n, p). We can see that all the curves have similar shape with maximum value around 0.135. This strengthens the conclusion we made before by inspecting the inequality (4). Another thing to notice is that when p increases with n, larger n indicates better performance, but when p is fixed the opposite happens. This indicates that the inequality, and therefore the guarantee for being able to separate outliers, holds better for large p. The reason for this is that, for fixed p, at most p points can be arranged in $\mathbb{R}^p$ such that all their pairwise Euclidean distances are equal. Hence, it is clear that larger p makes it easier for the approximation $\mathbf{D} \approx \tilde{\mathbf{D}}$ to hold.

Another regime in which $\|\mathbf{D}-\tilde{\mathbf{D}}\|_{\text{op}}$ can be expected to be small is when the two groups, the bulk and the outliers, both exhibit comparatively small within-group variations. Next, we demonstrate this scenario by formulating general conditions under

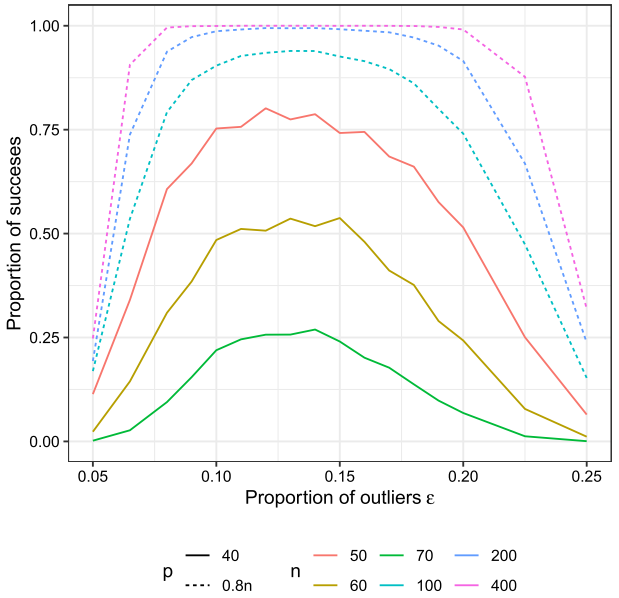

Fig. 1 Simulation results for Theorem 5

which the inequality (4) is asymptotically valid. For this, let $(\Omega, \mathcal{F}, P)$ be a probability space and let $(\mathcal{X}, d)$ be a complete and separable metric space. Throughout the remainder of this section, all probability distributions are assumed to be measurable w.r.t the Borel sets of $\mathcal{X}$. The following result quantifies the assumption of small within-group variability by assuming that the expected squared distance between two bulk elements goes to zero at a suitable rate, and similarly for the outliers.

Theorem 6 *Let $X_1, \ldots, X_{n_1}$ and $Y_1, \ldots, Y_{n_2}$ be mutually independent sets of i.i.d. random elements in $\mathcal{X}$, where $n_1/n \to 1-\varepsilon$, $n_2/n \to \varepsilon$, $\varepsilon \in (0, 1/2)$ and $n := n_1+n_2$. Assume further that the following hold,*

$$E\{d(X_1, X_2)^2\} = a_n, \quad E\{d(Y_1, Y_2)^2\} = b_n \quad \text{and} \quad E\{d(X_1, Y_1)\} \to c,$$

for some $a_n, b_n > 0$ such that $a_n, b_n = o(n^{-1})$ and some $c > 0$. Then, letting $\boldsymbol{D}$ be the $n \times n$ distance matrix of the sample $(X_1, \ldots, X_{n_1}, Y_1, \ldots, Y_{n_2})$, the probability that (4) holds goes to 1 as $n \to \infty$.

The scope of Theorem 6 is very wide as it applies to data residing in any metric space and fulfilling the above requirements. And while the assumption of vanishing within-group variation is somewhat impractical, on an empirical level it represents a scenario where the within-group variation is much smaller than the between-group variation.

3.4 Convergence of the weights

Two natural questions when using Algorithm 1 are that (i) how many iterations N should one use and, perhaps more importantly, (ii) whether some form of convergence occurs when N is allowed to grow. The latter questions have been discussed for the linear counterpart of our method, i.e., k-step W-estimation, under some particular weight functions, see Corollary 2.2 in Tyler (1987) and also the discussion in Taskinen et al. (2010). However, these classical results typically require that the observed set of n points is sufficiently irregularly spread in $\mathbb{R}^p$ and that $p \ll n$, see Condition 2.1(v) in Tyler (1987). In contrast, in our scenario, the data can arise from any metric space (or feature space, in case of kernel data), meaning that the classical results apply only if the data are embeddable into a low-dimensional Euclidean space in a sufficiently irregular way. As there is no guarantee that this is the case in general, other proof techniques need to be sought. We focus here on a simplified scenario where no centering (via $\mathbf{H}_n$) is applied and the weight function is taken to be $g(x) = x$. This choice was made as both the centering and the possibility of a general weight function make the iteration formula highly non-linear, leading to intractable analyses. Note that, while the following discussion concerns the convergence of the weights w_i, their convergence actually implies the convergence of the corresponding embedding too, since it is uniquely determined by the weights.

Using the notation of Sect. 3.2, let $\mathbf{U} \in \mathbb{R}^{n\times p}$ denote the matrix of the first p eigenvectors of $\mathbf{A}$, taken to be fixed in the following, and denote its rows by $\mathbf{u}_1, \ldots, \mathbf{u}_n \in \mathbb{R}^p$. Let $\mathbb{S}^{n-1}$ denote the unit simplex in $\mathbb{R}^n$. Inspection of the proof of Theorem 3 reveals that, in our simplified setting, the weights are updated by iterating the map $f_{\mathbf{U}} : \mathbb{S}^{n-1} \to \mathbb{S}^{n-1}$, defined as,

$$f_{\mathbf{U}}(\mathbf{w}) = \frac{\operatorname{diag}(\mathbf{U}\{\mathbf{U}^\top \operatorname{diag}(\mathbf{w})\mathbf{U} + \delta \mathbf{I}_p\}^{-1}\mathbf{U}^\top)}{\operatorname{tr}\{(\mathbf{U}^\top \operatorname{diag}(\mathbf{w})\mathbf{U} + \delta \mathbf{I}_p)^{-1}\}}, \tag{5}$$

where the regularization by a small value $\delta > 0$, for which Theorem 7 below suggests a minimum value, has been included to ensure that $f_{\mathbf{U}}$ is well-defined also on the boundary of the simplex $\mathbb{S}^{n-1}$. This regularization is precisely what allows us to circumvent regularity assumptions such as Condition 2.1(v) in Tyler (1987). Note that while the same regularization could also be incorporated in Algorithm 1 to ensure the invertibility of $\mathbf{U}^\top\{\operatorname{diag}(\mathbf{w}) - \mathbf{w}\mathbf{w}^\top\}\mathbf{U}$, our experiments revealed that this is not necessary in practice, see the simulation at the end of this section. Hence, regularization is not included in Algorithm 1. By the classical Banach fixed point theorem, a sufficient condition for the convergence of the iteration $\mathbf{w} \mapsto f_{\mathbf{U}}(\mathbf{w})$ (to some $\mathbf{w}$ for which $f_{\mathbf{U}}(\mathbf{w}) = \mathbf{w}$) is that the map $f_{\mathbf{U}}$ is *contractive*, i.e., that there exists $C \in (0, 1)$ such that for all $\mathbf{w}_1, \mathbf{w}_2 \in \mathbb{S}^{n-1}$ we have $\|f_{\mathbf{U}}(\mathbf{w}_1) - f_{\mathbf{U}}(\mathbf{w}_2)\| \leq C\|\mathbf{w}_1 - \mathbf{w}_2\|$, where $\|\cdot\|$ is some norm. The following result now gives a sufficient condition for the contractiveness to hold in case of the ℓ_1-norm, $\|\cdot\|_1$.

Theorem 7 *Let f_U be as in (5). Then, for all $\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathbb{S}^{n-1}$, we have*

$$\|f_U(\boldsymbol{w}_1) - f_U(\boldsymbol{w}_2)\|_1 \leq \frac{2\max_i(\|\boldsymbol{u}_i\|^2)}{\delta}\|\boldsymbol{w}_1 - \boldsymbol{w}_2\|_1.$$

By Theorem 7, a sufficient condition for the convergence of the weight iteration is that $\delta > 2\max_i(\|\mathbf{u}_i\|^2)$. This result gives us insight on which factors play a role in determining the possibility of convergence:

(i) Since the norm of $\mathbf{U}$ is a constant not depending on n, increasing n has on average a decreasing effect on $\max_i(\|\mathbf{u}_i\|^2)$. Consequently, the convergence is faster and less regularization is needed when the sample size n is large.
(ii) The dimensionality p of the embedding has the opposite effect and, on average, greater p equates with larger $\max_i(\|\mathbf{u}_i\|^2)$. Thus, convergence is the fastest for small-dimensional embeddings.
(iii) The term $\max_i(\|\mathbf{u}_i\|^2)$ tells the squared distance of the farthest standardized embedded coordinate from the origin, meaning that convergence is more difficult to achieve in the presence of highly outlying points. While somewhat counterintuitive, this is entirely in line with the classical theory of robust statistics, where methods start to show degenerate behavior when the data are concentrated on a subspace, which is approximately the case when the data consists of two sufficiently distant groups (bulk and outliers).

While the Banach fixed point theorem works with any choice of norm, possibly better constants in Theorem 7 could still be obtained by replacing the ℓ_1-norm with some other choice. Note also that Theorem 7 depends on the data solely through the matrix $\mathbf{U}$ (data embedding), and hence does not make any assumptions about the exact form of the original data (be it Euclidean or something else).

We next investigate the convergence of weights with a simulation study, using the fully general version of Algorithm 1 with centering and a weight function of the form $g(x) = x^\alpha$, $\alpha = -0.75, -0.25, 0.25, 0.75$. We carry this out by simulating various data sets of size n, running Algorithm 1 for them and recording how many iterations are required for the convergence of the weights in each case. We take as the definition of convergence that $\|\mathbf{w}_{k+1} - \mathbf{w}_k\|_1/n < 10^{-4}$, for two consecutive weight vectors $\mathbf{w}_k$ and $\mathbf{w}_{k+1}$. As our setting, we take data residing on the unit sphere in q-dimensional Euclidean space, generated as $\mathbf{x}_i/\|\mathbf{x}_i\|$ where $\mathbf{x}_i \sim \mathcal{N}_q(\lambda\mathbf{1}_q, \mathbf{I}_q)$. The proportion of outliers was taken to be $\varepsilon = 0.10$ and the parameter values $\lambda = 5$ and $\lambda = -5$ were used for the bulk and outliers, respectively. The choices $q = 25, 50$ were considered for the dimension of the extrinsic space and we took the sample sizes to be $n = 100, 200, 400$. We then used the arc length distance to compute the matrices $\mathbf{A}$ and $\mathbf{U} \in \mathbb{R}^{n\times p}$ where $p = 5, 10, 15$. To summarize, the simulation has a total of four parameters, n, p, q, α, and we ran 1000 replicates of the simulation for each of their combinations.

The results are reported in Fig. 2 in the form of stacked barplots. The colouring corresponds to a discretized version of the number of iterations required for convergence, e.g., the fastest convergence was observed in the settings with the bars of the lightest color. If no convergence was reached, we terminated the algorithm at iteration $N = 101$, and inspection of the raw results shows that this value was recorded only

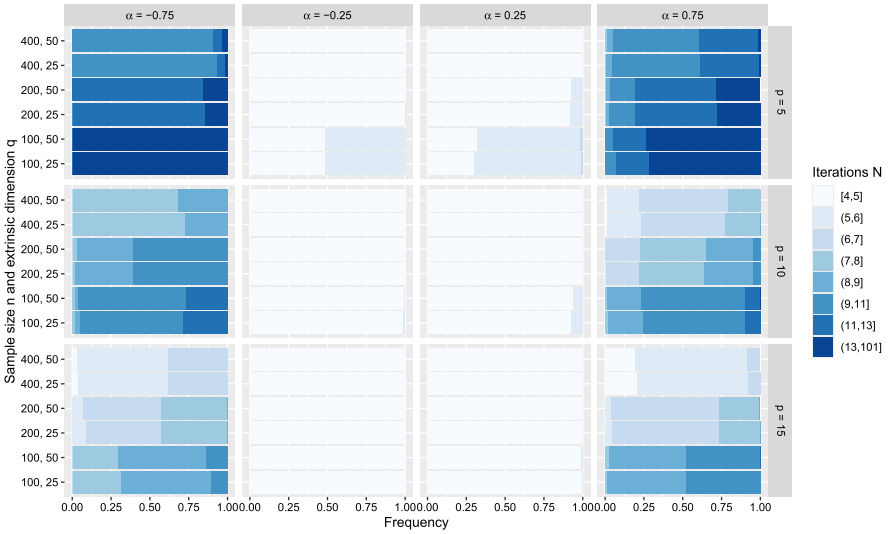

Fig. 2 Barplots showing the frequencies of the numbers of iterations N that were required for convergence in Algorithm 1 in various scenarios

once over the 72000 replicates. Thus, the results imply that convergence is essentially guaranteed in all the considered cases.

Next, we interpret the effects of each of the parameters in turn: Increasing n makes the convergence faster, an effect which we predicted already earlier by Theorem 7. The extrinsic dimension q has practically no effect on the convergence, which is as expected considering that the method does not operate on the original data but with its p-dimensional embedding. For α, only its absolute value seems to matter, with larger absolute values leading to slower convergence. This is again reasonable, since the trivial case $\alpha = 0$ (no weighting) corresponds to the fastest possible convergence. Finally, the most interesting finding is that larger p equates with faster convergence, which is opposite to what we inferred from Theorem 7. Our experiments (not shown here) revealed that this effect is caused by the centering, which is included in the simulation but not in Theorem 7. Indeed, it turns out that the centering, which is applied to the rows of $\mathbf{U}$, smooths out the distribution of the $\|\mathbf{u}_i\|^2$ suitably to cancel the adversary effect of p. Due to the reasons explained earlier, a careful analysis of this phenomenon appears highly non-trivial, and we leave it for future work.

The above results allow us to formulate some general guidelines regarding the choice of N in practice. In general, a small N appears sufficient but if one uses either a weight function $g(x) = x^\alpha$ with large $|\alpha|$ or has a small embedding dimension p, then additional iterations should be included. In any case, Algorithm 1 is relatively fast, meaning that any extra iterations incur only a minimal cost, and one might always want to run it until convergence. Finally, we note that we also redid the simulation with data generated from the multivariate normal and using the Euclidean distance but the results were qualitatively similar (not shown here).

4 Examples

4.1 Comparison to other outlier detection methods

We present a comparison between our proposed method and two other well-known outlier detection methods KNNo (Ramaswamy et al. 2000) and LOF (Breunig et al. 2000) on three datasets, WDBC (Wolberg et al. 1993), Parkinsons (Little 2007) and Spambase (Hopkins et al. 1999), all of which contain the ground truth information about outlying points. The comparison is based on the review paper Campos et al. (2016) where the authors of that article highlight that these datasets (among many) could be suitable for benchmarking outlier detection methods. They also claim that KNNo and LOF have great overall performance (and thus are good to compare against). The evaluation criterion, known as $P@n$, that we use in this study also comes from the same article. This criterion, which is essentially equivalent to the concept of *precision* in classification, calculates how large proportion of the k observations flagged to be outliers (where k is chosen to be the true number of outliers in the data) are actually outliers.

The first dataset WDBC consists of different features extracted for tissue samples and our goal is to differentiate the samples with malign changes from the benign ones. Parkinsons dataset has features extracted for voice samples and the goal is to find the samples having the Parkinsons disease. Spambase has features extracted from emails (for example, wordcounts) and our objective is detect spam (outliers). Regarding the competing methods, KNNo compares the distance of a point with its K nearest neighbors and declares those with highest distances to be outliers. Local outlier factor (LOF) compares the density (of other points) around a point with the densities of its neighbors and finds outliers that have unusually low densities. LOF involves a tuning parameter minPts quantifying the size of the local neighborhoods of the points. To summarize, when applied to the datasets, all three methods (KNNo, LOF and WKKP) thus produce a set of the k observations they deem the most outlying.

As a preprocessing step, we reduce the numbers of outliers in the data by taking only a random sample of them. For WDBC we have 357 observations of bulk and downsample to 20 outliers (from 212), for Parkinsons we have 48 bulk and downsample to 20 outliers (from 147) and for Spambase we downsample to 400 bulk (from 2788) and 50 outliers (from 1813). This downsampling allows us to perform multiple replicates of the following experiment by repeating the sampling 1000 times with different random seeds, each time fitting the models and calculating $P@n$. By aggregating over the 1000 repeats, we then compute the mean and standard deviation of $P@n$ for each combination of the method and the data. Several different parameter configurations are compared for all the methods, see the rows of Table 1. For WKPP, we used its kernel data version with the quadratic kernel and weight functions of the form $g(x) = x^{\alpha}$. For $\alpha > 0$ ($\alpha < 0$), the k observations with the largest (smallest) weights were taken to be outliers.

In Table 1 we show the mean values of $P@n$ along with the corresponding standard deviations in parentheses. We see that on WDBC all methods perform quite well with $P@n$ being around 0.75 and KNNo with $K = 5$ having the best performance of 0.764.

Table 1 Comparison of our proposed method to two other outlier methods (KNNo and LOF) on three datasets

		WDBC	Parkinsons	Spambase
WKPP	$\alpha = 0.5$, $p = 2$	0.605 (0.10)	0.162 (0.05)	0.168 (0.05)
	$\alpha = 0.5$, $p = 6$	0.664 (0.07)	0.188 (0.07)	0.345 (0.08)
	$\alpha = -0.5$, $p = 2$	0.754 (0.07)	0.081 (0.05)	0.334 (0.06)
	$\alpha = -0.5$, $p = 6$	0.751 (0.08)	0.186 (0.07)	0.358 (0.06)
KNNo	$K = 3$	0.761 (0.09)	0.437 (0.08)	0.345 (0.05)
	$K = 5$	0.764 (0.08)	0.449 (0.09)	0.337 (0.06)
	$K = 7$	0.763 (0.08)	0.428 (0.08)	0.326 (0.06)
LOF	MinPts = 5	0.217 (0.08)	0.390 (0.08)	0.131 (0.04)
	MinPts = 10	0.365 (0.11)	0.478 (0.08)	0.155 (0.05)
	MinPts = 20	0.735 (0.06)	0.485 (0.08)	0.181 (0.05)

Means of $P@n$ with standard deviations in parentheses

For Spambase, WKPP and KNNo both perform equally well with $P@n$ around 0.35 and WKPP manages to achieve the single best score of 0.358. We also observe that, for both data, WKPP performs better when α is taken to be negative (down-weighting outliers), instead of positive, and that for $\alpha < 0$ the two choices of p do not differ too much.

For the Parkinsons dataset, KNNo and LOF seem to perform the best and LOF with the parameter value MinPts $= 20$ has the best performance $P@n = 0.485$. Based on these results, it appears that WKPP does not work well in this case but, when investigating the reasons for this, we found out that if one takes in WKPP the smallest weights instead of largest and vice versa (i.e. we search for bulk instead of outliers), very good results are obtained. For example, for the parameters $\alpha = -0.5$, $p = 2$ the corresponding result is $P@n = 0.506$ (0.08), which actually outperforms the other methods. This phenomenon is well-known in the robust statistics literature: if the outlier group is sufficiently large (20 out of 68 observations here), then the detection methods might start to hold it as the bulk and the bulk as outliers. From a mathematical viewpoint, the threshold for this should be 50% of data, but this can be much smaller if the bulk consists of several subgroups, which is likely to be the case here. Note that a similar behavior was not observed with the other datasets as they have much more uneven proportions of bulk/outliers.

4.2 Image data example

We next apply WKPP to the Street View House Numbers (SVHN) data, available at http://ufldl.stanford.edu/housenumbers/, see also Netzer et al. (2011). The data consist of RGB-images of house number signs of size 32×32, of which we take only the first $n = 400$. The original purpose behind the data set is to classify the images according to the digits (0-9) they represent. However, some of the figures are badly cropped,

showing also parts of adjacent digits, or have poor image quality, implying that it is advisable to carry out an outlier detection step before the actual analysis, to get rid of any anomalous data points that might impair the classification. Hence, in this demonstration we apply the outlier detection capabilities of WKPP to the data, with the purpose of filtering out any irregular images.

As images are highly redundant objects (several pixels could typically be removed without sacrificing any information), a standard approach in image analysis is to replace the original images with a set of features that aggregate the pixel-level information in some meaningful way. Thus, as a preprocessing step, we compute a total of five features: average amount of the red, blue and green colors, along with the average horizontal and vertical Sobel filters which are commonly used in edge detection and are essentially discrete derivatives. We use the filtering matrix

$$\begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$$

for the horizontal filter and its transpose for the vertical filter. Moreover, to obtain a more comprehensive view of the images, we do not apply the features to the full images, but instead divide each of them to 64 subimages of size 4×4 and compute the features separately for each subimage. Finally, for each original image, we compute the correlation matrix of the five features over its 64 subimages. Hence, after this process, each original image $\mathbf{X}_i$ has been converted into 5×5 correlation matrix $\mathbf{S}_i$ describing the dependencies between the features in the image. Similar pre-processing was used, for example, in Alavi et al. (2013). Our chosen features allow us to discover images with a lot of color variation (color features) and a lot of variation in the presence of edges (Sobel features), making these features a natural choice in the current context of colorful number signs.

To apply WKPP, we still need to compute the pair-wise distance matrix between the correlation matrices $\mathbf{S}_i$ and for this we use the affine invariant Riemannian metric $d_{\mathrm{A}}(\mathbf{S}_i, \mathbf{S}_j) = \|\mathrm{Log}(\mathbf{S}_i^{-1/2}\mathbf{S}_j\mathbf{S}_i^{-1/2})\|_F$, where $\mathrm{Log}(\cdot)$ is the matrix logarithm. The metric d_{A} can be seen as the natural, geometry-acknowledging measure of distance in the positive-definite manifold where the correlation matrices reside (Bhatia 2009). Next, we apply WKPP to the resulting matrix $\mathbf{D}$ of squared distances as described in Sect. 3.2, using the dimension parameter $p \in \{1, \ldots, 30\}$, the weight function $g(x) = x$ and $N = 10$ iterations (we tried also with $N = 50$ but the results were almost identical). As a result, we obtain the final iterated weights, denoted in the following by O_{1i}, $i \in \{1, \ldots, n\}$. Since we use an increasing weight function, the outlying images are characterized with large values of O_{1i}. As a competing method, and a sort of "golden standard", we apply the metric lens depth studied in Cholaquidis et al. (2023); Geenens et al. (2023) which assigns each image its "depth", a measure of centrality. Since centrality and outlyingness are opposite quantities, we still translate the produced set of depths D_i into measures of outlyingness simply as $O_{2i} := 1 - D_i$. We note that, in general, lens depth and WKPP have quite different use cases. Lens

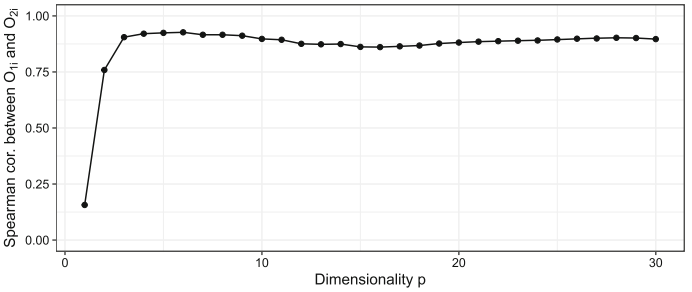

Fig. 3 Spearman correlation between the outlyingnesses O_{1i} and O_{2i} as a function of the dimension p

depth tells us how typical an object is with respect to a reference population (sample), by studying how the object in question is located w.r.t. randomly chosen pairs of objects from the population, see Cholaquidis et al. (2023); Geenens et al. (2023). In contrast, WKPP aims to embed the objects into a Euclidean space in a manner that down-weights deviating objects. Hence, it is difficult to predict in advance how their results should compare. However, it is intuitively clear that WKPP can be expected to work well if the outliers are separated from the bulk in geometry of the embedding space, whereas for lens depth they should be distinguishable in the geometry of the original metric space.

Before looking at the individual images, we first inspect Fig. 3 showing the plot of the Spearman correlation between the points O_{1i}, O_{2i} as a function of the dimension p. The plot reveals that, as soon as $p \geq 3$, both WKPP and the metric lens depth identify almost identical outlyingness structure in the data. As the metric lens depth is based on the geometric properties of the sample of correlation matrices (in the positive-definite manifold), this correspondence demonstrates that our weights, which are not inherently geometric in nature, are nevertheless able to capture geometrically meaningful structures in the data. Moreover, this behavior is not limited to smaller p but, as Fig. 3 demonstrates, applies to all values of p that one might reasonably consider in such a problem. Hence, the choice of p is not critical in this example, as long as it is sufficiently large, see Sect. 5 for more discussion on the selection of p in a general situation.

We next did a small timing study, comparing the running times of WKPP, as described in Algorithm 1, versus the metric lens depth, as described in formula (4.1) in Geenens et al. (2023). For WKPP, we considered several different values of p, each with $N = 10$ iterations. Each method was used to compute the outlyingnesses of the $n = 400$ images a total of 20 times. The experiments were performed on a desktop computer with AMD Ryzen 5 3600 6-core processor and 16 GB RAM.

The average times (in seconds) per method are shown in Table 2 and indicate that the choice of p has very little effect on the running speed of WKPP. This is expected as the main computational burden in Algorithm 1 comes from the extraction of p first eigenvectors of the $n \times n$ matrix $\mathbf{A}$ and the multiplications involved in computing $\mathbf{b}$ on each iteration, giving the method a complexity of $\mathcal{O}(n^2 pN)$. The complexity of the metric lens depth is simply $\mathcal{O}(n^3)$, giving WKPP a clear edge that is visible also in

Table 2 Average running times (in seconds) of the two methods for quantifying outlyingness over 20 replications. The columns 2-7 correspond to WKPP with different values of p, whereas the final column gives the computation time of the metric lens depth

p	5	10	15	20	25	30	Depth
Average time (s)	0.458	0.488	0.498	0.543	0.549	0.580	11.956

Table 2. Granted, there is the possibility of improving upon the direct implementation in formula (4.1) in Geenens et al. (2023), but it is not obvious how this could be achieved.

We show in Figs. 4 and 5 the 18 images with the highest and lowest, respectively, outlyingness values O_{1i} when $p = 3$. The most outlying images clearly differ from the ones with a small value of O_{1i}, exhibiting brighter colors and greater contrast between the digits and the background. We also see that several of the outliers belong to the same street signs (for example, the second and fifth images on the first row and the second image on the third row of Fig. 4 all correspond to the same sign displaying the number 251). Hence, to summarize, our proposed method was able to extract visually meaningful outliers that agree with the results of a more geometrically-aware method, but with significantly lower computational effort.

Finally, we remark that from a statistical viewpoint, the size of a sub-image used in the pre-processing is essentially a tuning parameter controlling a trade-off between globality and accuracy: for small sub-images, the information is highly local but the large number of sub-images allows us to accurately measure their variation and vice versa for small amount of sub-images. To explore this, we repeated the experiment with sub-images of size 8×8 and the resulting Spearman correlations turned out to be very similar (not shown here). Out of the 18 most outlying images found by using 8×8 and 4×4 sub-images, a total of 12 images were found by both. We also tried extracting the 50 most outlying images and in this case the overlap was 35, showing that, in this case, the exact choice of this tuning parameter does not appear to be too crucial.

5 Discussion

We proposed a method for embedding and outlier detection for metric space and kernel data. Increased robustness is achieved with iterative weighting of observations. This method can be used for outlier detection. We showed two examples: one where we compare WKPP to its competitors using three different real data sets and one where we identified outliers in image data.

One important question is how to choose the dimension parameter p, especially in the outlier detection context. Taking too small p increases bias, whereas large values of p are expected to increase variance. However, in the image example in Sect. 4 we saw that overestimating p was less critical than taking it too small. In metric space and kernel data there is, in general, no "true" number of (original) variables, but one

Fig. 4 Observations with the largest values O_{1i}

Fig. 5 Observations with the smallest values O_{1i}

could still envision using some information criterion or bootstrap to compare different numbers of components p. For example, bootstrap sampling could be used to identify eigenvalues of $\mathbf{A}$ which significantly differ from the later eigenvalues.

In Algorithm 1 we held the dimensionality p constant for all iterations, but one could also change its value between the iterations of the algorithm. This could be done so that one starts with large p and reduces the value step-by-step to some final value, which is used in the projection. The idea behind this is that we start with multiple directions in order not to lose relevant information and gradually lower p when the signal gets more and more concentrated to the leading directions. The idea of varying p along the iterations is the most natural in a non-Euclidean case, where there is no clear concept of dimensionality and no particular value of p holds special meaning.

As a third possible avenue for future research, one could further investigate the properties of Algorithm 1 in various scenarios. In Theorem 7 we established its convergence under particular well-behaving cases, but this still leaves open questions such as the speed or monotonicity of the convergence.

When one has obtained the final weights, in order to flag the outliers, a cut-off needs to be chosen and this can be done, for example, by either of the following ways: (a) Selecting a quantile level β, computing the β-quantile of the n weights under the assumption of a symmetric Dirichlet distribution for them and then flagging all points having more extreme weight than this quantile. This approach is similar in spirit to statistical null hypothesis testing. Under the null hypothesis that there are no outliers, all weights are expected to be equal on average and it seems reasonable to model them as being a realization from a symmetric Dirichlet distribution. If some observed weights are significantly outside the main support of this distribution, they provide evidence against the null hypothesis and we deem them as outliers. (b) Drawing a plot of the weights in descending order and looking for a point of evident drop. This is the same idea as in using scree plot for identifying the number of principal components. (c) Choosing directly a percentage α and flagging the $\lfloor n\alpha \rfloor$ observations with the most extreme weights as outlying. Regardless of the cut-off method one uses, closer observing of the flagged data is recommended, as outlier detection should always be complemented with the visual inspection of the data. We leave the closer study of these procedures to future work.

Appendix A: Maximal invariants and kernelization

In this section, we review the concept of maximal invariant and provide the necessary background for proving Theorem 1.

First, we recall the definitions of invariant and maximal invariant statistics (Lehmann and Romano 2005).

Definition 2 (invariant statistics) Let $\mathbb{X}$ be a set and G be a group of one-to-one measurable functions $g : \mathbb{X} \to \mathbb{X}$. Let $t : \mathbb{X} \to \mathcal{X}$ be a statistic taking values in some space $\mathcal{X}$. We say that t is an *invariant statistic under* G, if $t(x) = t(g(x))$ for all $g \in G$ and $x \in \mathbb{X}$. If, in addition to that, $t(x) = t(y)$ implies $x = g(y)$ for some $g \in G$, we say that t is a *maximal invariant under* G.

Next, we present (Lehmann and Romano 2005, Theorem 6.2.1) which gives us the reason to prove that the inner product matrix is a maximal invariant under orthogonal transformations.

Theorem 8 *Let* M *be a maximal invariant under* G. *Then, a necessary and sufficient condition for* t *to be invariant is that it depends on* x *only through* M(x)*; that is, that there exists a function* $h : \mathcal{X} \to \mathcal{X}$ *for which* $t(x) = h\{M(x)\}$ *for all* $x \in \mathbb{X}$.

Theorem 9 *The statistic* $t : \mathbb{R}^{n\times p} \to \mathbb{R}^{n\times n}$ *defined as* $t(\boldsymbol{X}) = \boldsymbol{X}\boldsymbol{X}^\top$ *is a maximal invariant under the transformations* $\boldsymbol{X} \mapsto \boldsymbol{X}\boldsymbol{V}$ *where* $\boldsymbol{V} \in \mathbb{R}^{p\times p}$ *is an orthogonal matrix.*

We can wrap up that the inner product matrix $\mathbf{X}\mathbf{X}^\top$ is a maximal invariant under orthogonal transformations, and therefore all those methods that are invariant under orthogonal transformations can be calculated only using $\mathbf{X}\mathbf{X}^\top$. Slightly modifying the proof of Theorem 9 now lets us obtain a version for double centered kernels.

Theorem 10 *The statistic* $t : \mathbb{R}^{n\times p} \to \mathbb{R}^{n\times n}$ *defined as* $t(\boldsymbol{X}) = \boldsymbol{HXX}^\top \boldsymbol{H}$ *is a maximal invariant under the transformations* $\boldsymbol{X} \mapsto \boldsymbol{XV} + \boldsymbol{1}_n \boldsymbol{b}^\top$ *where* $\boldsymbol{V} \in \mathbb{R}^{p\times p}$ *is an orthogonal matrix and* $\boldsymbol{b} \in \mathbb{R}^p$.

Theorem 1 in the main text now follows by combining Theorems 8 and 10.

Finally, as an application of Theorem 1, we next show that the predictions produced by ridge regression with any penalization parameter $\lambda \geq 0$ satisfy the conditions of Theorem 1.

$$
\begin{aligned}
\hat{\mathbf{y}} =& \mathbf{H}\mathbf{X}\hat{\boldsymbol{\beta}}^R = \mathbf{H}\mathbf{X}(\mathbf{X}^\top\mathbf{H}\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{X}^\top\mathbf{H}\mathbf{y} \\
\mapsto& \mathbf{H}(\mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top)\{(\mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top)^\top\mathbf{H}(\mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top) + \lambda\mathbf{I}_p\}^{-1} \\
& (\mathbf{X}\mathbf{V} + \mathbf{1}_n\mathbf{b}^\top)^\top\mathbf{H}\mathbf{y} \\
=& \mathbf{H}\mathbf{X}\mathbf{V}(\mathbf{V}^\top\mathbf{X}^\top\mathbf{H}\mathbf{X}\mathbf{V} + \lambda\mathbf{I}_p)^{-1}\mathbf{V}^\top\mathbf{X}^\top\mathbf{H}\mathbf{y} \\
=& \mathbf{H}\mathbf{X}\mathbf{V}(\mathbf{V}^\top\mathbf{X}^\top\mathbf{H}\mathbf{X}\mathbf{V} + \lambda\mathbf{V}^\top\mathbf{V})^{-1}\mathbf{V}^\top\mathbf{X}^\top\mathbf{H}\mathbf{y} \\
=& \mathbf{H}\mathbf{X}\mathbf{V}(\mathbf{V}^\top(\mathbf{X}^\top\mathbf{H}\mathbf{X} + \lambda\mathbf{I}_p)\mathbf{V})^{-1}\mathbf{V}^\top\mathbf{X}^\top\mathbf{H}\mathbf{y} \\
=& \mathbf{H}\mathbf{X}\mathbf{V}\mathbf{V}^\top(\mathbf{X}^\top\mathbf{H}\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{V}\mathbf{V}^\top\mathbf{X}^\top\mathbf{H}\mathbf{y} \\
=& \mathbf{H}\mathbf{X}(\mathbf{X}^\top\mathbf{H}\mathbf{X} + \lambda\mathbf{I}_p)^{-1}\mathbf{X}^\top\mathbf{H}\mathbf{y} \\
=& \hat{\mathbf{y}}.
\end{aligned}
$$

Appendix B: Proofs

Proof of Theorem 9 We are going to prove that $\mathbf{X}\mathbf{X}^\top = \mathbf{Y}\mathbf{Y}^\top$ if and only if $\mathbf{Y} = \mathbf{X}\mathbf{V}^\top$ for some orthogonal matrix $\mathbf{V}$. Now clearly, if $\mathbf{Y} = \mathbf{X}\mathbf{V}^\top$, then $\mathbf{Y}\mathbf{Y}^\top = \mathbf{X}\mathbf{V}^\top\mathbf{V}\mathbf{X}^\top = \mathbf{X}\mathbf{X}^\top$.

Let us now assume that $\mathbf{X}\mathbf{X}^\top = \mathbf{Y}\mathbf{Y}^\top$ and that the matrices have singular value decompositions $\mathbf{X} = \mathbf{U}_1\mathbf{D}_1\mathbf{V}_1^\top$ and $\mathbf{Y} = \mathbf{U}_2\mathbf{D}_2\mathbf{V}_2^\top$. Now we cut the matrices so that $\mathbf{U}_i \in \mathbb{R}^{n\times r}$ and $\mathbf{D}_i \in \mathbb{R}^{r\times r}$ and $\mathbf{V}_i \in \mathbb{R}^{r\times p}$, where $r = \operatorname{rank}(\mathbf{X})$ (we see $\operatorname{rank}(\mathbf{X}) = \operatorname{rank}(\mathbf{X}\mathbf{X}^\top) = \operatorname{rank}(\mathbf{Y}\mathbf{Y}^\top) = \operatorname{rank}(\mathbf{Y})$) or the number of strictly positive singular values. Then we have $\mathbf{X}\mathbf{X}^\top = \mathbf{Y}\mathbf{Y}^\top \implies \mathbf{U}_1\mathbf{D}_1^2\mathbf{U}_1^\top = \mathbf{U}_2\mathbf{D}_2^2\mathbf{U}_2^\top$. Because eigenvalues are unique and singular values positive, we get $\mathbf{D}_1 = \mathbf{D}_2$. Let the eigenvalues in matrix $\mathbf{D}_1^2$ be $\lambda_1, \ldots, \lambda_m$ with (algebraic) multiplicities $a_1, \ldots, a_m$ ($\sum a_i = r$). Now we have that the relation between two sets of eigenvectors, $\mathbf{U}_1$ and $\mathbf{U}_2$, of the same matrix, is $\mathbf{U}_2 = \mathbf{U}_1\mathbf{W}^\top$, where

$$
\mathbf{W} = \begin{pmatrix} \mathbf{W}_1 & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_2 & \ldots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \ldots & \mathbf{W}_m \end{pmatrix},
$$

in which $\mathbf{W}_i \in \mathbb{R}^{r_i \times r_i}$ are orthogonal blocks (Horn and Johnson 2013, Theorem 2.5.4). The equation we are interested in now becomes

$$\begin{aligned}\mathbf{Y} &= \mathbf{U}_2\mathbf{D}_2\mathbf{V}_2^\top \\ &= \mathbf{U}_2\mathbf{D}_1\mathbf{V}_2^\top \\ &= \mathbf{U}_1\mathbf{W}^\top\mathbf{D}_1\mathbf{V}_2^\top \\ &= \mathbf{U}_1\mathbf{D}_1\mathbf{W}^\top\mathbf{V}_2^\top \\ &= \mathbf{U}_1\mathbf{D}_1\mathbf{V}_1^\top\mathbf{V}_1\mathbf{W}^\top\mathbf{V}_2^\top \\ &= \mathbf{X}\mathbf{V}^\top,\end{aligned}$$

where $\mathbf{V} = \mathbf{V}_2\mathbf{W}\mathbf{V}_1^\top$ is orthogonal. □

Proof of Theorem 10 The start of the proof is equivalent to Theorem 9, but instead of $\mathbf{X}$ and $\mathbf{Y}$ we consider $\mathbf{HX}$ and $\mathbf{HY}$. By doing that, we end up with the statement $\mathbf{HY} = \mathbf{HXV}^\top$.

Now, because the nullspace of the matrix $\mathbf{H}$ is $\{c\mathbf{1}_n \mid c \in \mathbb{R}\}$, we have that $\mathbf{HY} = \mathbf{HXV}^\top \implies \mathbf{H}(\mathbf{Y} - \mathbf{XV}^\top) = \mathbf{0}$, implying that $\mathbf{Y} - \mathbf{XV}^\top = \mathbf{1}_n\mathbf{b}^\top \implies \mathbf{Y} = \mathbf{XV}^\top + \mathbf{1}_n\mathbf{b}^\top$ for some $\mathbf{b} \in \mathbb{R}^p$. □

Proof of Theorem 2 Let $\mathbf{V} \in \mathbb{R}^{p\times p}$ be an orthogonal matrix, $\mathbf{S}(\mathbf{X}) = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ the eigendecomposition of the scatter matrix $\mathbf{S}(\mathbf{X})$ and $\mathbf{Z}(\mathbf{X}) = \mathbf{HXU}$ the matrix of principal component scores based on $\mathbf{S}(\mathbf{X})$. When we replace $\mathbf{X}$ by $\mathbf{XV}^\top + \mathbf{1}_n\mathbf{b}^\top$, the eigendecomposition becomes $\mathbf{S}(\mathbf{XV}^\top + \mathbf{1}_n\mathbf{b}^\top) = \mathbf{V}\mathbf{S}(\mathbf{X})\mathbf{V}^\top = \mathbf{VU}\mathbf{\Lambda}\mathbf{U}^\top\mathbf{V}^\top$. Because $\mathbf{VU}$ is orthogonal, it consists of the eigenvectors of $\mathbf{S}(\mathbf{XV}^\top + \mathbf{1}_n\mathbf{b}^\top)$ whereas the eigenvalues are the same as for $\mathbf{X}$. Now $\mathbf{Z}(\mathbf{XV}^\top + \mathbf{1}_n\mathbf{b}^\top) = \mathbf{H}(\mathbf{XV}^\top + \mathbf{1}_n\mathbf{b}^\top)\mathbf{VU} = \mathbf{HXU} = \mathbf{Z}(\mathbf{X})$ and therefore the principal component scores are invariant under orthogonal transformations. □

Proof of Theorem 3 By definition, the mean can be calculated $\bar{\mathbf{x}}_{k+1} = \mathbf{X}^\top\mathbf{w}_k$. Letting then $\mathbf{X}_k = \mathbf{H}_k^\top\mathbf{X}$ be the data matrix with the columns of $\mathbf{X}$ centered using the mean vector $\bar{\mathbf{x}}_{k+1}$, we have the expression,

$$\mathbf{w}_{k+1} = \frac{1}{\mathrm{tr}[g\{\mathrm{diag}(\mathbf{X}_k\mathbf{S}_{k+1}^{-1}\mathbf{X}_k^\top)\}]}g\{\mathrm{diag}(\mathbf{X}_k\mathbf{S}_{k+1}^{-1}\mathbf{X}_k^\top)\}, \tag{B1}$$

where g is calculated componentwise.

Since $\mathbf{H}_k^\top\mathbf{H} = \mathbf{H}_k^\top$, we have $\mathbf{X}_k = \mathbf{H}_k^\top\mathbf{HX}$, giving,

$$\begin{aligned}\mathbf{S}_{k+1} &= \mathbf{X}_k^\top\mathrm{diag}(\mathbf{w}_k)\mathbf{X}_k \\ &= (\mathbf{HX})^\top\mathbf{H}_k\mathrm{diag}(\mathbf{w}_k)\mathbf{H}_k^\top(\mathbf{HX}) \\ &= (\mathbf{HX})^\top(\mathbf{I}_n - \mathbf{w}_k\mathbf{1}_n^\top)\mathrm{diag}(\mathbf{w}_k)\mathbf{H}_k^\top(\mathbf{HX}) \\ &= (\mathbf{HX})^\top\{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k\mathbf{1}_n^\top\mathrm{diag}(\mathbf{w}_k)\}\mathbf{H}_k^\top(\mathbf{HX})\end{aligned}$$

$$
\begin{aligned}
&= (\mathbf{HX})^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top\} \mathbf{H}_k^\top (\mathbf{HX}) \\
&= (\mathbf{HX})^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top\} (\mathbf{I}_n - \mathbf{1}_n \mathbf{w}_k^\top)(\mathbf{HX}) \\
&= (\mathbf{HX})^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top - \mathrm{diag}(\mathbf{w}_k)\mathbf{1}_n \mathbf{w}_k^\top \\
&\quad + \mathbf{w}_k \mathbf{w}_k^\top \mathbf{1}_n \mathbf{w}_k^\top\} (\mathbf{HX}) \\
&= (\mathbf{HX})^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top - \mathbf{w}_k \mathbf{w}_k^\top + \mathbf{w}_k \mathbf{w}_k^\top\} (\mathbf{HX}) \\
&= (\mathbf{HX})^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top\} (\mathbf{HX}). \qquad \text{(B2)}
\end{aligned}
$$

By using the relations $\mathbf{H}_k^\top \mathbf{H} = \mathbf{H}_k^\top$ and $\mathbf{HXX}^\top \mathbf{H} = \mathbf{A} = \mathbf{U}\mathbf{\Pi}^2\mathbf{U}^\top$, we observe that $\mathbf{HX}$ has the SVD $\mathbf{HX} = \mathbf{U}\mathbf{\Pi}\mathbf{V}^\top$, where $\mathbf{V}$ is a $p \times p$ orthogonal matrix. Using this and (B2), we obtain

$$
\begin{aligned}
\mathbf{X}_k \mathbf{S}_{k+1}^{-1} \mathbf{X}_k^\top &= \mathbf{H}_k^\top (\mathbf{HX}) \{(\mathbf{HX})^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top\} (\mathbf{HX})\}^{-1} \\
&\quad (\mathbf{HX})^\top \mathbf{H}_k \\
&= \mathbf{H}_k^\top (\mathbf{U}\mathbf{\Pi}\mathbf{V}^\top) [(\mathbf{U}\mathbf{\Pi}\mathbf{V}^\top)^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top\} \\
&\quad (\mathbf{U}\mathbf{\Pi}\mathbf{V}^\top)]^{-1} (\mathbf{U}\mathbf{\Pi}\mathbf{V}^\top)^\top \mathbf{H}_k \\
&= \mathbf{H}_k^\top \mathbf{U} [\mathbf{U}^\top \{\mathrm{diag}(\mathbf{w}_k) - \mathbf{w}_k \mathbf{w}_k^\top\} \mathbf{U}]^{-1} \mathbf{U}^\top \mathbf{H}_k.
\end{aligned}
$$

Plugging in to (B1) we now get the desired result. □

Proof of Theorem 4 Direct calculations reveal, after some algebra, that the ith element of the vector $a(\mathbf{x})$ is $(\mathbf{x}_i - \bar{\mathbf{x}})^\top (\mathbf{x} - \bar{\mathbf{x}})$.

As $\mathbf{A}$ has the eigendecomposition $\mathbf{U}\mathbf{\Pi}^2\mathbf{U}^\top$, then $\mathbf{HX}$ has the SVD $\mathbf{U}\mathbf{\Pi}\mathbf{V}^\top$, where $\mathbf{V}$ is a $p \times p$ orthogonal matrix. Now, by the proof of Theorem 3, the w_i-weighted scatter matrix of the training sample can be written as $\mathbf{S}_0 := \mathbf{X}^\top \{\mathrm{diag}(\mathbf{w}) - \mathbf{w}\mathbf{w}^\top\} \mathbf{X} = (\mathbf{HX})^\top \{\mathbf{H}_k \mathrm{diag}(\mathbf{w}) \mathbf{H}_k^\top\} (\mathbf{HX})$. Plugging now in $\mathbf{HX} = \mathbf{U}\mathbf{\Pi}\mathbf{V}^\top$ and the definition of $\mathbf{M}$ from the theorem statement reveals that the matrix $\mathbf{VM}$ contains the eigenvectors of $\mathbf{S}_0$ as its columns. Consequently, the centered principal component scores of $\mathbf{x} \in \mathbb{R}^p$ equal

$$
\begin{aligned}
&(\mathbf{VM})^\top \left\{ \mathbf{x} - \bar{\mathbf{x}} - \sum_{i=1}^n w_i (\mathbf{x}_i - \bar{\mathbf{x}}) \right\} \\
&\quad = \mathbf{M}^\top \mathbf{\Pi}^{-1} \mathbf{U}^\top \mathbf{U}\mathbf{\Pi}\mathbf{V}^\top (\mathbf{x} - \bar{\mathbf{x}}) - \mathbf{M}^\top \mathbf{V}^\top \mathbf{V}\mathbf{\Pi}\mathbf{U}^\top \mathbf{w} \\
&\quad = \mathbf{M}^\top \mathbf{\Pi}^{-1} \mathbf{U}^\top a(\mathbf{x}) - \mathbf{M}^\top \mathbf{\Pi}\mathbf{U}^\top \mathbf{w},
\end{aligned}
$$

establishing the claim. □

Proof of Corollary 1 As the jth element of the vector $a(\mathbf{x})$ is $(\mathbf{x}_j - \bar{\mathbf{x}})^\top (\mathbf{x} - \bar{\mathbf{x}})$, we see that $a(\mathbf{x}_i)$ for a training point $\mathbf{x}_i$ is the ith column of the matrix $\mathbf{A}$. By using that and writing (3) for matrices, where columns correspond to points, we get

$$
\mathbf{M}^\top \mathbf{\Pi}^{-1} \mathbf{U}^\top \mathbf{A} - \mathbf{M}^\top \mathbf{\Pi}\mathbf{U}^\top \mathbf{w} \mathbf{1}_n^\top
$$

$$
\begin{aligned}
&= \mathbf{M}^\top \mathbf{\Pi}^{-1} \mathbf{U}^\top \mathbf{U} \mathbf{\Pi}^2 \mathbf{U}^\top - \mathbf{M}^\top \mathbf{\Pi} \mathbf{U}^\top \mathbf{w} \mathbf{1}_n^\top \\
&= \mathbf{M}^\top \mathbf{\Pi} \mathbf{U}^\top - \mathbf{M}^\top \mathbf{\Pi} \mathbf{U}^\top \mathbf{w} \mathbf{1}_n^\top \\
&= \mathbf{M}^\top \mathbf{\Pi} \mathbf{U}^\top (\mathbf{I}_n - \mathbf{w} \mathbf{1}_n^\top) \\
&= \mathbf{M}^\top \mathbf{\Pi} \mathbf{U}^\top \mathbf{H}_k
\end{aligned}
$$

Now when we transpose this to have observations as rows, we get $\mathbf{Z} = \mathbf{H}_k^\top \mathbf{U} \mathbf{\Pi} \mathbf{M}$. □

Proof of Theorem 5 Throughout the proof, we denote $\mathbf{H} = \mathbf{I}_n - (1/n)\mathbf{1}_n \mathbf{1}_n^\top$. We first show that the desired separation holds if and only if

$$
\max\{|u_1|, \ldots, |u_{n_1}|\} < \min\{|u_{n_1+1}|, \ldots, |u_n|\}, \tag{B3}
$$

where $\mathbf{u} = (u_1, \ldots, u_n)^\top$ is a first eigenvector of the matrix $\mathbf{A} = (-1/2)\mathbf{HDH}$. By Algorithm 1, the weight w_i is obtained as

$$
w_i = \frac{g([\mathbf{H}\mathbf{u}\mathbf{u}^\top \mathbf{H}]_{ii})}{\sum_{j=1}^n g([\mathbf{H}\mathbf{u}\mathbf{u}^\top \mathbf{H}]_{jj})}.
$$

As g is strictly increasing, the weights w_i are separated (in the sense claimed in the statement of the theorem) if and only if the elements $[\mathbf{H}\mathbf{u}\mathbf{u}^\top \mathbf{H}]_{ii}$ are separated. Now, $\mathbf{A}\mathbf{1}_n = 0$, implying that $\mathbf{1}_n \in \mathbb{R}^n$ is an eigenvector of $\mathbf{A}$. If the eigenvalue α corresponding to the leading eigenvector $\mathbf{u}$ of $\mathbf{A}$ has $\alpha \leq 0$, then the matrix $\mathbf{A} = (-1/2)\mathbf{HDH}$ would be negative semi-definite. This, in turn, would mean that

$$
0 \leq \operatorname{tr}(\mathbf{HDH}) = -\frac{1}{n}\mathbf{1}_n^\top \mathbf{D}\mathbf{1}_n,
$$

which can only happen in the trivial case of $\mathbf{D} = \mathbf{0}$, which we disallowed. Hence, $\alpha > 0$, and $\mathbf{u}$ is orthogonal to the vector $\mathbf{1}_n$ corresponding to the eigenvalue 0. Consequently, $\mathbf{H}\mathbf{u}\mathbf{u}^\top \mathbf{H} = \mathbf{u}\mathbf{u}^\top$ and the elements $[\mathbf{H}\mathbf{u}\mathbf{u}^\top \mathbf{H}]_{ii}$ are separated if and only if u_i^2 are separated which is equivalent to (B3).

Now, to see that (B3) holds under our assumptions, we observe that, by Lemma 3, we have

$$
\left\| \mathbf{u} - \frac{1}{\sqrt{n}} \begin{pmatrix} \sqrt{\frac{n_2}{n_1}} \mathbf{1}_{n_1} \\ -\sqrt{\frac{n_1}{n_2}} \mathbf{1}_{n_2} \end{pmatrix} \right\| \leq \frac{\sqrt{2}n \|\mathbf{D} - \tilde{\mathbf{D}}\|_{\mathrm{op}}}{n_1 n_2 (\bar{d}_{12} - \frac{\bar{d}_{11} + \bar{d}_{22}}{2})}.
$$

(Note that, technically u might have its sign flipped, but this case can be treated in the same way, and hence we assume that the sign is as above.) Now, Lemma 2 says that, as soon as, $n_2 > n_1$ and

$$
\frac{\sqrt{2}n \|\mathbf{D} - \tilde{\mathbf{D}}\|_{\mathrm{op}}}{n_1 n_2 (\bar{d}_{12} - \frac{\bar{d}_{11} + \bar{d}_{22}}{2})} \leq \frac{\sqrt{\frac{n_2}{nn_1}} - \sqrt{\frac{n_1}{nn_2}}}{2},
$$

we have

$$\min\{|u_1|, \ldots, |u_{n_1}|\} > \max\{|u_{n_1+1}|, \ldots, |u_n|\}.$$

The claimed result now follows by applying the above in a scenario where the two groups (outliers and bulk) are flipped the other way around (i.e., by essentially exchanging n_1 and n_2, in which case the above separation takes the form (B3)). □

Lemma 2 *Let $a > 0$ and $b < 0$ be such that $|a| > |b|$. Let $\boldsymbol{r} \in \mathbb{R}^{n_1}$ and $\boldsymbol{s} \in \mathbb{R}^{n_2}$ be such that*

$$\left\| \begin{pmatrix} \boldsymbol{r} \\ \boldsymbol{s} \end{pmatrix} - \begin{pmatrix} a\boldsymbol{1}_{n_1} \\ b\boldsymbol{1}_{n_2} \end{pmatrix} \right\| \leq \frac{a+b}{2}.$$

Then,

$$\min\{|r_1|, \ldots, |r_{n_1}|\} > \max\{|s_1|, \ldots, |s_{n_2}|\}.$$

Proof of Lemma 2 Let us define $\varepsilon = (a+b)/2$. Now we take a look at the ε-environments of a and b and how they are transformed under $x \mapsto x^2$. If the squared ε-environments overlap, then $(b-\varepsilon)^2 > (a-\varepsilon)^2 \implies \varepsilon > (a+b)/2$ which is not possible by the assumption. Therefore, assume that the squared ε-environments do not overlap.

By the assumption and by equivalence of norms we have

$$\begin{aligned} &\max\{|r_1 - a_1|, \ldots, |r_{n_1} - a_{n_1}|, \\ &|s_1 - b_1|, \ldots, |s_{n_2} - b_{n_2}|\} \\ =&\left\| \begin{pmatrix} \mathbf{r} \\ \mathbf{s} \end{pmatrix} - \begin{pmatrix} a\mathbf{1}_{n_1} \\ b\mathbf{1}_{n_2} \end{pmatrix} \right\|_\infty \leq \left\| \begin{pmatrix} \mathbf{r} \\ \mathbf{s} \end{pmatrix} - \begin{pmatrix} a\mathbf{1}_{n_1} \\ b\mathbf{1}_{n_2} \end{pmatrix} \right\|_2 \leq \varepsilon. \end{aligned}$$

This says that all the components $r_i \in [a-\varepsilon, a+\varepsilon]$ and $s_i \in [b-\varepsilon, b+\varepsilon]$. Because the squared ε-environments do not overlap, we have

$$\begin{aligned} &\min\{r_1^2, \ldots, r_{n_1}^2\} > \max\{s_1^2, \ldots, s_{n_2}^2\} \\ \implies &\min\{|r_1|, \ldots, |r_{n_1}|\} > \max\{|s_1|, \ldots, |s_{n_2}|\}. \end{aligned}$$

□

Lemma 3 *Let $\boldsymbol{T}$ be the symmetric $n \times n$ block matrix*

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^\top & \boldsymbol{C} \end{pmatrix},$$

where the diagonal blocks are square and have the sizes $n_1 \times n_1$ and $n_2 \times n_2$, respectively. Denote by $\bar{a} := \boldsymbol{1}_{n_1}^\top \boldsymbol{A} \boldsymbol{1}_{n_1}/n_1^2$, $\bar{b} := \boldsymbol{1}_{n_1}^\top \boldsymbol{B} \boldsymbol{1}_{n_2}/(n_1 n_2)$, $\bar{c} := \boldsymbol{1}_{n_2}^\top \boldsymbol{C} \boldsymbol{1}_{n_2}/n_2^2$ the means

of the three blocks of $\boldsymbol{T}$ *and let* $\tilde{\boldsymbol{T}}$ *be the block matrix*

$$\tilde{\boldsymbol{T}} := \begin{pmatrix} \bar{a}\boldsymbol{J}_{n_1,n_1} & \bar{b}\boldsymbol{J}_{n_1,n_2} \\ \bar{b}\boldsymbol{J}_{n_2,n_1} & \bar{c}\boldsymbol{J}_{n_2,n_2} \end{pmatrix},$$

where $\boldsymbol{J}_{n_1,n_2}$ *denotes a matrix of size* $n_1 \times n_2$ *full of ones. Assume that* $\bar{a}$, $\bar{b}$, $\bar{c} > 0$ *and that* $\bar{b} > \max\{\bar{a}, \bar{c}\}$

Let $\boldsymbol{u} \in \mathbb{R}^n$ *denote an arbitrary leading eigenvector of the matrix* $(-1/2)\boldsymbol{HTH}$ *whose sign is chosen such that* $\boldsymbol{u}^\top \boldsymbol{t} \geq 0$ *where*

$$\boldsymbol{t} := \frac{1}{\sqrt{n}} \begin{pmatrix} \sqrt{\frac{n_2}{n_1}}\boldsymbol{1}_{n_1} \\ -\sqrt{\frac{n_1}{n_2}}\boldsymbol{1}_{n_2} \end{pmatrix}.$$

Then,

$$\|\boldsymbol{u} - \boldsymbol{t}\| \leq \frac{\sqrt{2}n\|\boldsymbol{T} - \tilde{\boldsymbol{T}}\|_{op}}{n_1 n_2(\bar{b} - \frac{\bar{a}+\bar{c}}{2})}.$$

Proof of Lemma 3 The matrix $\tilde{\mathbf{T}}$ can be written as

$$\tilde{\mathbf{T}} = \mathbf{v}\mathbf{v}^\top - \mathbf{w}\mathbf{w}^\top,$$

where

$$\mathbf{v} = \frac{1}{\sqrt{\bar{a}}} \begin{pmatrix} \bar{a}\mathbf{1}_{n_1} \\ \bar{b}\mathbf{1}_{n_2} \end{pmatrix} \quad \text{and} \quad \mathbf{w} = \frac{1}{\sqrt{\bar{a}}} \begin{pmatrix} \mathbf{0}_{n_1} \\ \sqrt{(\bar{b})^2 - \bar{a}\bar{c}}\mathbf{1}_{n_2} \end{pmatrix},$$

and both $\mathbf{v}$ and $\mathbf{w}$ are well-defined quantities by our assumptions on $\bar{a}$, $\bar{b}$, $\bar{c}$. Denote next $\theta := \sqrt{(\bar{b})^2 - \bar{a}\bar{c}}/(\bar{b} - \bar{a})$. Then, $\mathbf{w} = \theta(\mathbf{v} - \sqrt{\bar{a}}\mathbf{1}_n)$ and $\tilde{\mathbf{T}}$ can further be written as

$$\tilde{\mathbf{T}} = (\mathbf{v} \mid \theta\mathbf{v} - \theta\sqrt{\bar{a}}\mathbf{1}_n) \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} (\mathbf{v} \mid \theta\mathbf{v} - \theta\sqrt{\bar{a}}\mathbf{1}_n)^\top.$$

Consequently, using the fact that $\mathbf{H}\mathbf{1}_n = \mathbf{0}$, we get

$$-\frac{1}{2}\mathbf{H}\tilde{\mathbf{T}}\mathbf{H} = -\frac{1-\theta^2}{2}\mathbf{H}\mathbf{v}(\mathbf{H}\mathbf{v})^\top.$$

This means that the matrix $(-1/2)\mathbf{H}\tilde{\mathbf{T}}\mathbf{H}$ has rank 1. Now,

$$\mathbf{H}\mathbf{v} = \frac{\bar{b} - \bar{a}}{\sqrt{\bar{a}}} \begin{pmatrix} -n_2/n\mathbf{1}_{n_1} \\ n_1/n\mathbf{1}_{n_2} \end{pmatrix},$$

meaning that $\|\mathbf{Hv}\|^2 = n_1 n_2(\bar{b} - \bar{a})^2/(n\bar{a})$. Thus, the leading eigenvalue ϕ of $-\frac{1}{2}\mathbf{H}\tilde{\mathbf{T}}\mathbf{H}$ is

$$\phi_= - \frac{1-\theta^2}{2}\|\mathbf{Hv}\|^2 = -\frac{1}{2n} n_1 n_2(\bar{a} - 2\bar{b} + \bar{c}),$$

and a corresponding unit length eigenvector $\mathbf{t}$ is

$$\mathbf{t} = \frac{1}{\sqrt{n}} \begin{pmatrix} \sqrt{\frac{n_2}{n_1}}\mathbf{1}_{n_1} \\ -\sqrt{\frac{n_1}{n_2}}\mathbf{1}_{n_2} \end{pmatrix}.$$

Note that, by our assumptions, $\bar{a} - 2\bar{b} + \bar{c}$ is a strictly negative quantity.

We next use the Davis–Kahan theorem, see, e.g., Corollary 1 in Yu et al. (2015), to approximate the leading eigenvector of the matrix $(-1/2)\mathbf{HTH}$ with the leading eigenvector of $(-1/2)\mathbf{H}\tilde{\mathbf{T}}\mathbf{H}$. Namely, let $\mathbf{u}$ be any leading eigenvector of $(-1/2)\mathbf{HTH}$ whose sign is chosen such that $\mathbf{u}^\top\mathbf{t} \geq 0$. Then, by Corollary 1 in Yu et al. (2015),

$$\|\mathbf{u} - \mathbf{t}\| \leq \frac{2^{3/2} n \|\mathbf{H}(\mathbf{T} - \tilde{\mathbf{T}})\mathbf{H}\|_{\mathrm{op}}}{n_1 n_2(2\bar{b} - \bar{a} - \bar{c})} \leq \frac{2^{3/2} n \|\mathbf{T} - \tilde{\mathbf{T}}\|_{\mathrm{op}}}{n_1 n_2(2\bar{b} - \bar{a} - \bar{c})},$$

where the second inequality follows since (i) $\mathbf{H}$ is an orthogonal projection, meaning that all its eigenvalues are either equal to zero or equal to one, and (ii) the operator norm is sub-multiplicative. □

Before proving Theorem 6, we first present three auxiliary lemmas.

Lemma 4 *Let X_1, Y_1 be independent random elements in $\mathcal{X}$. Then,*

$$Var\{d(X_1, Y_1)\} \leq ([E\{d(X_1, X_2)^2\}]^{1/2} + [E\{d(Y_1, Y_2)^2\}]^{1/2})^2$$

where (X_2, Y_2) is an independent copy of (X_1, Y_1).

Proof of Lemma 4 Triangle inequality and the reverse triangle inequality give

$$\begin{aligned} |d(X_1, Y_1) - d(X_2, Y_2)| &\leq |d(X_1, Y_1) - d(X_1, Y_2)| + |d(X_1, Y_2) - d(X_2, Y_2)| \\ &\leq d(X_1, X_2) + d(Y_1, Y_2). \end{aligned}$$

Consequently,

$$\begin{aligned} \mathrm{Var}\{d(X_1, Y_1)\} &= \mathrm{E}[\{d(X_1, Y_1) - d(X_2, Y_2)\}^2] \\ &\leq \mathrm{E}\{d(X_1, X_2)^2\} + 2\mathrm{E}\{d(X_1, X_2)\}\mathrm{E}\{d(Y_1, Y_2)\} + \mathrm{E}\{d(Y_1, Y_2)^2\}, \end{aligned}$$

from which the claim follows using the power mean inequality. □

Lemma 5 *Let $X_1, \ldots, X_n$ be i.i.d. random elements in $\mathcal{X}$ and let $\boldsymbol{D} = \{d(X_i, X_j)\} \in \mathbb{R}^{n\times n}$ and $\bar{d} := \boldsymbol{1}_n'\boldsymbol{D}\boldsymbol{1}_n/n^2$. Then,*

$$E(\|\boldsymbol{D} - \bar{d}\boldsymbol{J}_{n,n}\|_F^2) \leq (2n-1)(n-1)E(d_{12}^2)$$

Proof of Lemma 5 We have

$$\mathrm{E}(\|\mathbf{D} - \bar{d}\mathbf{J}_{n,n}\|_F^2) = \sum_{i\neq j}^{n} \mathrm{E}\{(d_{ij} - \bar{d})^2\} + n\mathrm{E}\{(\bar{d}\,)^2\}.$$

This simplifies to

$$\begin{aligned}
&n(n-1)\mathrm{E}(d_{12}^2) - 2\sum_{i\neq j}^{n}\mathrm{E}(d_{ij}\bar{d}) + n(n-1)\mathrm{E}\{(\bar{d}\,)^2\} + n\mathrm{E}\{(\bar{d}\,)^2\}\\
=&n(n-1)\mathrm{E}(d_{12}^2) - 2n^{-2}\sum_{i\neq j}^{n}\sum_{k\neq \ell}^{n}\mathrm{E}(d_{ij}d_{k\ell}) + n^{-2}\sum_{i\neq j}^{n}\sum_{k\neq \ell}^{n}\mathrm{E}(d_{ij}d_{k\ell})\\
\leq&n(n-1)\mathrm{E}(d_{12}^2) + n^{-2}n^2(n-1)^2\mathrm{E}(d_{12}^2)\\
=&(2n-1)(n-1)\mathrm{E}(d_{12}^2),
\end{aligned}$$

where the inequality uses Cauchy-Schwarz. □

Lemma 6 *Let $X_1, \ldots, X_{n_1}$ and $Y_1, \ldots, Y_{n_2}$ be mutually independent sets of i.i.d. random elements in $\mathcal{X}$ and let $\boldsymbol{D} = \{d(X_i, Y_j)\} \in \mathbb{R}^{n_1\times n_2}$ and $\bar{d} := \boldsymbol{1}_{n_1}'\boldsymbol{D}\boldsymbol{1}_{n_2}/(n_1n_2)$. Then,*

$$E(\|\boldsymbol{D} - \bar{d}\boldsymbol{J}_{n_1,n_2}\|_F^2) \leq 3n_1n_2Var(d_{11}).$$

Proof of Lemma 5 We have

$$\mathrm{E}(\|\mathbf{D} - \bar{d}\mathbf{J}_{n_1,n_2}\|_F^2) = \sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\Big[\mathrm{Var}(d_{ij}) + \mathrm{E}\{(d_{ij}-\mu)(\mu-\bar{d})\} + \mathrm{E}\{(\mu-\bar{d})^2\}\Big],$$

where $\mu := \mathrm{E}(d_{12})$. Using Cauchy-Schwarz, we then get the upper bound

$$\mathrm{E}(\|\mathbf{D} - \bar{d}\mathbf{J}_{n_1,n_2}\|_F^2) \leq n_1n_2\mathrm{Var}(d_{11}) + n_1n_2\sqrt{\mathrm{Var}(d_{11})}\sqrt{\mathrm{Var}(\bar{d})} + n_1n_2\mathrm{Var}(\bar{d}).$$

Cauchy-Schwarz gives $\mathrm{Var}(\bar{d}) \leq \mathrm{Var}(d_{11})$, after which plugging in into the above yields the claim. □

Proof of Theorem 6 As $\bar{d}_{12}$ is a 2nd degree U-statistic, by our assumptions, $\bar{d}_{12} - (\bar{d}_{11} + \bar{d}_{22})/2$ converges to the positive constant c. Hence, the claim follows

once we show that

$$\{\mathrm{E}(\|\mathbf{D}-\tilde{\mathbf{D}}\|_{\mathrm{op}})\}^2 = o(n).$$

We begin by bounding the left-hand side as

$$\begin{aligned}
&\{\mathrm{E}(\|\mathbf{D}-\tilde{\mathbf{D}}\|_{\mathrm{op}})\}^2 \\
\leq&\mathrm{E}(\|\mathbf{D}-\tilde{\mathbf{D}}\|_{\mathrm{op}}^2) \\
\leq&\mathrm{E}(\|\mathbf{D}_{11}-\bar{d}_{11}\mathbf{J}_{n_1,n_1}\|_{\mathrm{F}}^2)+2\mathrm{E}(\|\mathbf{D}_{12}-\bar{d}_{12}\mathbf{J}_{n_1,n_2}\|_{\mathrm{F}}^2)+\mathrm{E}(\|\mathbf{D}_{22}-\bar{d}_{22}\mathbf{J}_{n_2,n_2}\|_{\mathrm{F}}^2).
\end{aligned}$$

Using Lemmas 4–6, this is further upper bounded by

$$\begin{aligned}
&(2n_1-1)(n_1-1)\mathrm{E}\{d(X_1,\,X_2)^2\}+(2n_2-1)(n_2-1)\mathrm{E}\{d(Y_1,\,Y_2)^2\} \\
&+6n_1n_2([\mathrm{E}\{d(X_1,\,X_2)^2\}]^{1/2}+[\mathrm{E}\{d(Y_1,\,Y_2)^2\}]^{1/2})^2,
\end{aligned}$$

from which the claim now follows. □

Proof of Theorem 7 Letting

$$A(\mathbf{w}) := \{\mathbf{U}^\top \mathrm{diag}(\mathbf{w})\mathbf{U}+\delta\mathbf{I}_p\}^{-1},$$

the left-hand side of the claim can be written as

$$\sum_{i=1}^{n}\left|\mathbf{u}_i^\top\left(\frac{A(\mathbf{w}_1)}{\|A(\mathbf{w}_1)\|_1}-\frac{A(\mathbf{w}_2)}{\|A(\mathbf{w}_2)\|_1}\right)\mathbf{u}_i\right| =: \sum_{i=1}^{n}|\mathbf{u}_i^\top\mathbf{G}\mathbf{u}_i|, \tag{B4}$$

where $\|\cdot\|_1$ denotes (for a matrix argument) the Schatten 1-norm (which is equal to the trace for symmetric positive semi-definite matrices).

Let now $\mathbf{D}\in\mathbb{R}^{n\times n}$ be diagonal matrix whose diagonal elements correspond to the signs of the diagonal elements of the matrix $\mathbf{U}\mathbf{G}\mathbf{U}^\top$. Then, (B4) equals $|\mathrm{tr}(\mathbf{D}\mathbf{U}\mathbf{G}\mathbf{U}^\top)|$ and we have,

$$\sum_{i=1}^{n}|\mathbf{u}_i^\top\mathbf{G}\mathbf{u}_i|\leq\sum_{i=1}^{n}\sigma_i(\mathbf{D})\sigma_i(\mathbf{U}\mathbf{G}\mathbf{U}^\top),$$

where $\sigma_i(\cdot)$ denotes the ith singular value and the inequality follows from Von Neumann trace inequality. As all singular values of $\mathbf{D}$ equal one, we thus have that

$$\begin{aligned}
&\|f_{\mathbf{U}}(\mathbf{w}_1)-f_{\mathbf{U}}(\mathbf{w}_2)\|_1 \\
\leq&\left\|\frac{A(\mathbf{w}_1)}{\|A(\mathbf{w}_1)\|_1}-\frac{A(\mathbf{w}_2)}{\|A(\mathbf{w}_2)\|_1}\right\|_1 \\
\leq&\left\|\frac{A(\mathbf{w}_1)}{\|A(\mathbf{w}_1)\|_1}-\frac{A(\mathbf{w}_2)}{\|A(\mathbf{w}_1)\|_1}\right\|_1
\end{aligned}$$

$$
\begin{aligned}
&+ \left\| \frac{A(\mathbf{w}_2)}{\|A(\mathbf{w}_1)\|_1} - \frac{A(\mathbf{w}_2)}{\|A(\mathbf{w}_2)\|_1} \right\|_1 \\
=&\frac{1}{\|A(\mathbf{w}_1)\|_1} \|A(\mathbf{w}_1) - A(\mathbf{w}_2)\|_1 \\
&+ \|A(\mathbf{w}_2)\|_1 \left| \frac{\|A(\mathbf{w}_1)\|_1 - \|A(\mathbf{w}_2)\|_1}{\|A(\mathbf{w}_1)\|_1 \|A(\mathbf{w}_2)\|_1} \right| \\
\leq&\frac{2}{\|A(\mathbf{w}_1)\|_1} \|A(\mathbf{w}_1) - A(\mathbf{w}_2)\|_1,
\end{aligned}
$$

where the second inequality uses triangle inequality and the third one reverse triangle inequality. Reversing the roles of $\mathbf{w}_1$ and $\mathbf{w}_2$ we get,

$$
\begin{aligned}
&\|f_{\mathbf{U}}(\mathbf{w}_1) - f_{\mathbf{U}}(\mathbf{w}_2)\|_1 \\
\leq&\frac{2}{\max\{\|A(\mathbf{w}_1)\|_1,\ \|A(\mathbf{w}_2)\|_1\}} \|A(\mathbf{w}_1) - A(\mathbf{w}_2)\|_1.
\end{aligned} \tag{B5}
$$

Now, using the decomposition $A(\mathbf{w}_1) - A(\mathbf{w}_2) = A(\mathbf{w}_2)\{A(\mathbf{w}_2)^{-1} - A(\mathbf{w}_1)^{-1}\}A(\mathbf{w}_1)$ and the Hölder inequality for the Schatten 1-norm, we get

$$
\begin{aligned}
&\|A(\mathbf{w}_1) - A(\mathbf{w}_2)\|_1 \\
\leq&\|A(\mathbf{w}_1)\|_\infty \|A(\mathbf{w}_2)\|_\infty \|A(\mathbf{w}_1)^{-1} - A(\mathbf{w}_2)^{-1}\|_1.
\end{aligned}
$$

Applying this to (B5), in the case $\|A(\mathbf{w}_1)\|_1 \geq \|A(\mathbf{w}_2)\|_1$ we get

$$
\begin{aligned}
&\|f_{\mathbf{U}}(\mathbf{w}_1) - f_{\mathbf{U}}(\mathbf{w}_2)\|_1 \\
\leq&2\|A(\mathbf{w}_2)\|_\infty \left\| A(\mathbf{w}_1)^{-1} - A(\mathbf{w}_2)^{-1} \right\|_1 \\
\leq&\frac{2}{\delta} \|A(\mathbf{w}_1)^{-1} - A(\mathbf{w}_2)^{-1}\|_1,
\end{aligned}
$$

where the second inequality holds since

$$
\|A(\mathbf{w}_2)\|_\infty = \sigma_p^{-1}(\mathbf{U}^\top \mathrm{diag}(\mathbf{w}_1)\mathbf{U} + \delta \mathbf{I}_p) \leq 1/\delta.
$$

The same bound is obtained analogously for the case $\|A(\mathbf{w}_1)\|_1 < \|A(\mathbf{w}_2)\|_1$. The desired claim thus follows once we show that

$$
\|A(\mathbf{w}_1)^{-1} - A(\mathbf{w}_2)^{-1}\|_1 \leq \max_i (\|\mathbf{u}_i\|^2) \|\mathbf{w}_1 - \mathbf{w}_2\|_1.
$$

To see this, we denote $\mathbf{Z} := \mathrm{diag}(\mathbf{w}_1) - \mathrm{diag}(\mathbf{w}_2)$ and write,

$$
\|A(\mathbf{w}_1)^{-1} - A(\mathbf{w}_2)^{-1}\|_1 = \|\mathbf{U}^\top \mathbf{Z} \mathbf{U}\|_1 = \left\| \sum_{i=1}^n z_{ii} \mathbf{u}_i \mathbf{u}_i^\top \right\|_1,
$$

which is by the triangle inequality bounded by

$$\sum_{i=1}^{n} |z_{ii}| \left\| \mathbf{u}_i \mathbf{u}_i^\top \right\|_1 \leq \left(\sum_{i=1}^{n} |z_{ii}| \right) \max_i (\|\mathbf{u}_i\|^2),$$

proving the claim. □

Acknowledgements The authors thank Matthijs Lof for excellent comments. We thank the Editor, Associate Editor and referees, as well as our financial sponsors.

Funding Open Access funding provided by University of Turku (including Turku University Central Hospital). The work of Lauri Heinonen was supported by the Research Council of Finland (Grants 321968 and 353769). The work of Henri Nyberg was supported by the Research Council of Finland (Grant 321968) and the Foundation for Economic Education (Liikesivistysrahasto, Grant 220246). The work of Joni Virta was supported by the Research Council of Finland (Grants 335077, 347501 and 353769).

Declarations

Conflict of interest The authors have no Conflict of interest to declare.

References

Alavi A, Harandi MT, Sanderson C (2013) Relational divergence based classification on Riemannian manifolds. In: IEEE workshop on applications of computer vision, pp. 111–116

Amagata D, Onizuka M, Hara T (2021) Fast and exact outlier detection in metric spaces: a proximity graph-based approach. In: proceedings of the 2021 international conference on management of data, pp. 36–48

Aronszajn N (1950) Theory of reproducing kernels. Trans Am Math Soc 68(3):337–404

Blouvshtein L, Cohen-Or D (2018) Outlier detection for robust multi-dimensional scaling. IEEE Trans Pattern Anal Mach Intell 41(9):2273–2279

Boser BE, Guyon IM, Vapnik VN (1992) A training algorithm for optimal margin classifiers. In: proceedings of the 5th annual workshop on computational learning theory, pp. 144–152

Bhatia R (2009) Positive definite matrices. Princeton University Press, Princeton

Breunig MM, Kriegel H-P, Ng RT, Sander J (2000) Lof: identifying density-based local outliers. In: proceedings of the 2000 ACM SIGMOD international conference on management of data, pp. 93–104

Bhattacharya R, Patrangenaru V (2003) Large sample theory of intrinsic and extrinsic sample means on manifolds. Ann Stat 31(1):1–29

Cayton L, Dasgupta S (2006) Robust Euclidean embedding. In: proceedings of the 23rd international conference on machine learning, pp. 169–176

Cholaquidis A, Fraiman R, Gamboa F, Moreno L (2023) Weighted lens depth: some applications to supervised classification. Canadian J Stat 51(2):652–673

Cristianini N, Shawe-Taylor J (2004) Kernel methods for pattern analysis, vol 173. Cambridge University Press, Cambridge

Cornea E, Zhu H, Kim P, Ibrahim JG (2017) Regression models on Riemannian symmetric spaces. J Royal Stat Soc. Ser B, Stat Methodol 79(2):463
Campos GO, Zimek A, Sander J, Campello RJ, Micenková B, Schubert E, Assent I, Houle ME (2016) On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study. Data Min Knowl Disc 30:891–927
Dai X, Lopez-Pintado S (2023) Tukey's depth for object data. J Am Stat Assoc 118(543):1760–1772
Dubey P, Müller H-G (2019) Fréchet analysis of variance for random objects. Biometrika 106(4):803–821
Duan X, Tian Z, Qi P, Liu X (2011) A robust weighted kernel principal component analysis algorithm. In: 2011 international conference of information technology, computer engineering and management sciences, vol. 1, pp. 267–270
Fischer D, Berro A, Nordhausen K, Ruiz-Gazen A (2021) Repplab: an R package for detecting clusters and outliers using exploratory projection pursuit. Commun Stat-Simul Comput 50(11):3397–3419
Forero PA, Giannakis GB (2012) Sparsity-exploiting robust multidimensional scaling. IEEE Trans Signal Process 60(8):4118–4134
Geenens G, Nieto-Reyes A, Francisci G (2023) Statistical depth in abstract metric spaces. Stat Comput 33(2):46
Horn RA, Johnson CR (2013) Matrix analysis, 2nd edn. Cambridge University Press, Cambridge
Hopkins M, Reeber E, Forman G, Suermondt J (1999) Spambase. UCI Machine Learning Repository. https://doi.org/10.24432/C53G6X
Hampel F, Ronchetti E, Rousseeuw P, Stahel W (1986) Robust statistics: the approach based on the influence functions. Wiley, New York
Hofmann T, Schölkopf B, Smola AJ (2008) Kernel methods in machine learning. Ann Stat 36(3):1171–1220
Huang S-Y, Yeh Y-R, Eguchi S (2009) Robust kernel principal component analysis. Neural Comput 21(11):3179–3213
Little M (2007) Parkinsons. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C59C74
Lehmann EL, Romano JP (2005) Testing statistical hypotheses. Springer, New York
Lee JA, Verleysen M (2007) Nonlinear dimensionality reduction. Springer, New York
Lyons R (2013) Distance covariance in metric spaces. Ann Probab 41(5):3284–3305
Mardia KV, Jupp PE (2000) Directional statistics. Wiley, Chichester
Mardia KV, Kent JT, Bibby JM (1979) Multivariate analysis. Academic Press, Amsterdam
Nordhausen K, Tyler DE (2015) A cautionary note on robust covariance plug-in methods. Biometrika 102(3):573–588
Netzer Y, Wang T, Coates A, Bissacco A, Wu B, Ng AY (2011) Reading digits in natural images with unsupervised feature learning. In: NIPS workshop on deep learning and unsupervised feature learning
Pawlowsky-Glahn V, Buccianti A (2011) Compositional data analysis. Wiley, Chichester
Petersen A, Müller H-G (2019) Fréchet regression for random objects with Euclidean predictors. Ann Stat 47(2):691–719
Reza MS, Ruhi S (2015) Multivariate outlier detection using independent component analysis. Sci J Appl Math Stat 3(171):01
Ramaswamy S, Rastogi R, Shim K (2000) Efficient algorithms for mining outliers from large data sets. In: proceedings of the 2000 ACM SIGMOD international conference on management of Data, pp. 427–438
Schölkopf B, Smola A, Müller K-R (1998) Nonlinear component analysis as a kernel eigenvalue problem. Neural Comput 10(5):1299–1319
Shawe-Taylor J, Cristianini N (2004) Kernel methods for pattern analysis. Cambridge University Press, Cambridge
Schreurs J, Vranckx I, Hubert M, Suykens JA, Rousseeuw PJ (2021) Outlier detection in non-elliptical data by kernel MRCD. Stat Comput 31(5):1–18
Tyler DE, Critchley F, Dümbgen L, Oja H (2009) Invariant co-ordinate selection. J Royal Stat Soc: Ser B (Stat Methodol) 71(3):549–592
Taskinen S, Sirkiä S, Oja H (2010) k-step shape estimators based on spatial signs and ranks. J Stat Plann Inference 140(11):3376–3388
Tyler DE (1987) A distribution-free M-estimator of multivariate scatter. Annal Stat, 234–251
Virta J, Lee K-Y, Li L (2022) Sliced inverse regression in metric spaces. Stat Sin 32:2315–2337
Wolberg W, Mangasarian O, Street N, Street W (1993) Breast cancer wisconsin (diagnostic). UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5DW2B
You K, Park H-J (2021) Re-visiting Riemannian geometry of symmetric positive definite matrices for the analysis of functional connectivity. Neuroimage 225:117464

Yu Y, Wang T, Samworth RJ (2015) A useful variant of the Davis-Kahan theorem for statisticians. Biometrika 102(2):315–323
Zuo Y, Serfling R (2000) General notions of statistical depth function. Ann Stat, 461–482
Zhou S, Xiu N, Qi H-D (2020) Robust Euclidean embedding via EDM optimization. Math Program Comput 12(3):337–387