Journal of Machine Learning Research 17 (2016) 1-30 Submitted 3/16; Revised 10/16; Published 12/16

Multi-scale Classification using Localized Spatial Depth

Subhajit Dutta DUTTAS@IITK.AC.IN
Department of Mathematics and Statistics
Indian Institute of Technology
Kanpur 208016, India.

Soham Sarkar SOHAMSARKAR1991@GMAIL.COM
Anil K. Ghosh AKGHOSH@ISICAL.AC.IN
Theoretical Statistics and Mathematics Unit
Indian Statistical Institute
203, B. T. Road, Kolkata 700108, India.

Editor: Jie Peng

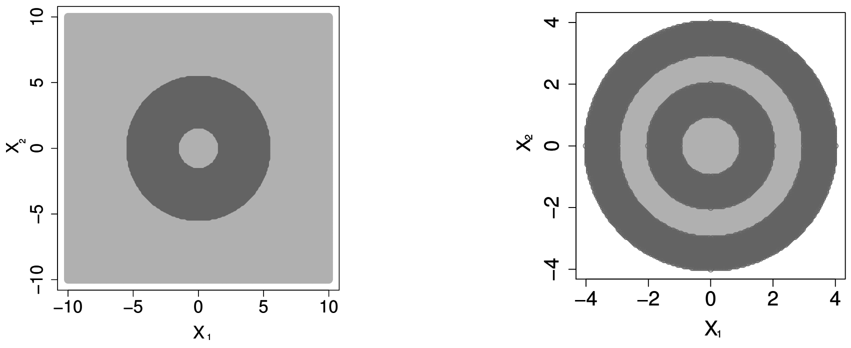
Abstract

In this article, we develop and investigate a new classifier based on features extracted using spatial depth. Our construction is based on fitting a generalized additive model to posterior probabilities of different competing classes. To cope with possible multi-modal as well as non-elliptic nature of the population distribution, we also develop a localized version of spatial depth and use that with varying degrees of localization to build the classifier. Final classification is done by aggregating several posterior probability estimates, each of which is obtained using this localized spatial depth with a fixed scale of localization. The proposed classifier can be conveniently used even when the dimension of the data is larger than the sample size, and its good discriminatory power for such data has been established using theoretical as well as numerical results.

Keywords: Bayes classifier, elliptic distributions, generalized additive models, HDLSS asymptotics, uniform strong consistency, weighted aggregation of posteriors.

1. Introduction

In a supervised classification problem with J competing classes, we have n_j labeled observations $\mathbf{x}_{j1}, \ldots, \mathbf{x}_{jn_j}$ from the j-th class $(1 \leq j \leq J)$. We use this training sample consisting of $n = \sum_{j=1}^{J} n_j$ observations to construct a decision rule for classifying an unlabeled observation $\mathbf{x}$ to one of these J classes. If π_j, f_j and $p(j|\cdot)$ denote the prior probability, the probability density function and the posterior probability of the j-th class, respectively, then the *Bayes classifier* assigns $\mathbf{x}$ to the class j_0, where $j_0 = \text{argmax}_{1 \leq j \leq J}\, p(j|\mathbf{x}) = \text{argmax}_{1 \leq j \leq J}\, \pi_j f_j(\mathbf{x})$. However, the f_j's or the $p(j|\cdot)$'s are usually unknown in practice, and one needs to estimate them from the training sample. Popular parametric classifiers like linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) (see, e.g., Hastie et al., 2009) are motivated by parametric model assumptions on the f_j's. So, they may lead to poor classification when these assumptions fail to hold, and the class boundaries of the Bayes classifier have complex geometry. On the other hand, nonparametric classifiers like those based on k-nearest neighbors (k-NN) (see, e.g., Cover and Hart, 1967) and kernel



(a) Example **E1**

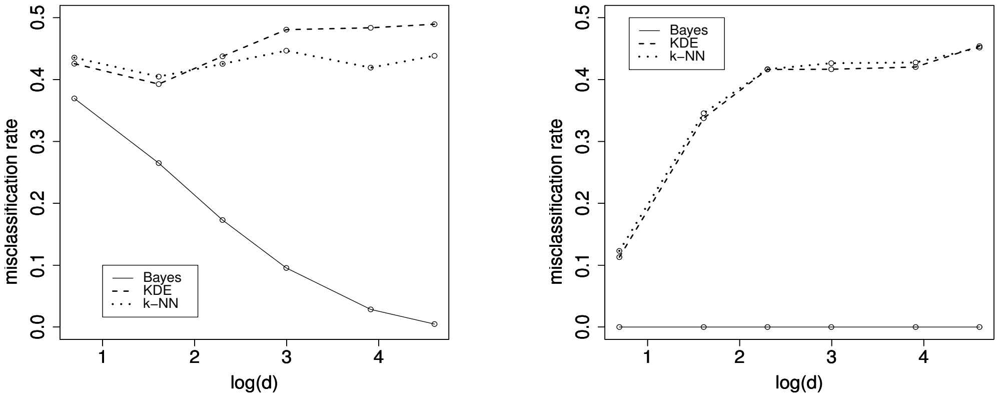
(b) Example **E2**

Figure 1: Bayes class boundaries in $\mathbb{R}^2$.

density estimates (KDE) (see, e.g., Scott, 2015) are more flexible and free from such model assumptions. But, they suffer from the curse of dimensionality and are often not suitable for high-dimensional data.

To demonstrate this, let us consider two examples denoted by **E1** and **E2**. **E1** involves a classification problem with two classes in $\mathbb{R}^d$, where the distribution of the first class is an equal mixture of $\mathrm{N}_d(\mathbf{0}_d, \mathbf{I}_d)$ and $\mathrm{N}_d(\mathbf{0}_d, 10\mathbf{I}_d)$, and that of the second class is $\mathrm{N}_d(\mathbf{0}_d, 5\mathbf{I}_d)$. Here N_d denotes the d-variate normal distribution, $\mathbf{0}_d = (0, \ldots, 0)^T \in \mathbb{R}^d$ and $\mathbf{I}_d$ is the $d \times d$ identity matrix. In **E2**, each class distribution is an equal mixture of two uniform distributions. For the first (respectively, the second) class, it is a mixture of $\mathrm{U}_d(0, 1)$ and $\mathrm{U}_d(2, 3)$ (respectively, $\mathrm{U}_d(1, 2)$ and $\mathrm{U}_d(3, 4)$), where $\mathrm{U}_d(r_1, r_2)$ denotes the uniform distribution over the region $\{\mathbf{x} \in \mathbb{R}^d : r_1 \leq \|\mathbf{x}\| \leq r_2\}$ with $0 \leq r_1 < r_2 < \infty$ and $\| \cdot \|$ being the Euclidean norm. Figure 1 shows the class boundaries of the Bayes classifier for these two examples when $d = 2$ and $\pi_1 = \pi_2 = 1/2$. The regions colored grey (respectively, black) correspond to observations classified to the first (respectively, the second) class by the Bayes classifier. It is clear that classifiers like LDA and QDA, or any other classifier with linear or quadratic class boundaries will deviate significantly from the Bayes classifier in both examples. A natural question then is how standard nonparametric classifiers like those based on k-NN and KDE perform in such examples.

Figure 2 shows the average misclassification rates of these two classifiers along with the Bayes risks for different values of d. These classifiers were trained on a sample of size 100 from each class, and the misclassification rates were computed based on 250 independent observations from each class. This procedure was repeated 500 times to calculate the average misclassification rates. Smoothing parameters associated with k-NN and KDE (i.e., the number of neighbors k in k-NN and the bandwidth in KDE) were chosen by minimizing leave-one-out cross-validation estimates of misclassification rates (see, e.g., Hastie et al., 2009). Figure 2 shows that in **E1**, the Bayes risk decreases to zero as d grows. Since the class distributions in **E2** have disjoint supports, the Bayes risk is zero for all values of d. But in both examples, the misclassification rates of these two nonparametric classifiers increased to almost 50% as d increased.



(a) Example **E1**

(b) Example **E2**

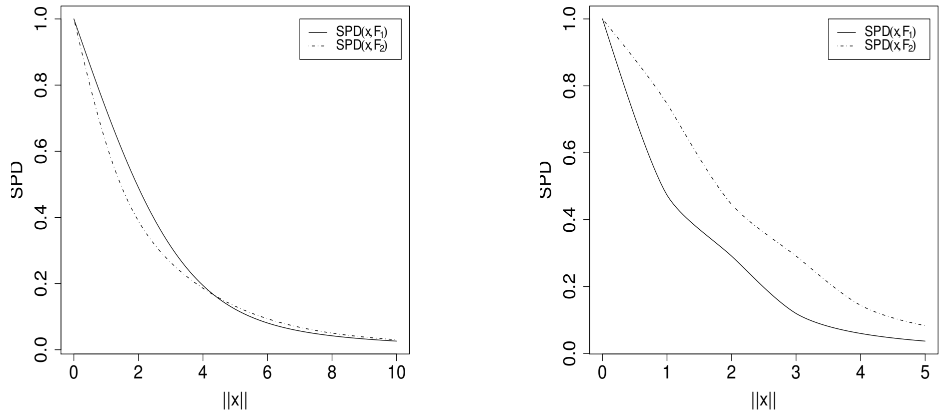
Figure 2: Average misclassification rates of nonparametric classifiers and the Bayes classifier for $d = 2, 5, 10, 20, 50$ and 100.

These two examples clearly show the necessity to develop new classifiers to cope with such situations. We use the idea of data depth for this purpose. Over the last three decades, data depth (see, e.g., Liu et al., 1999; Zuo and Serfling, 2000) has emerged as a powerful tool for multivariate data analysis with applications in many areas including supervised and unsupervised classification (see, e.g., Jornsten, 2004; Ghosh and Chaudhuri, 2005a,b; Hoberg and Mosler, 2006; Xia et al., 2008; Dutta and Ghosh, 2012; Li et al., 2012; Lange et al., 2014; Paindaveine and Van Bever, 2015). Spatial depth (also known as the L_1 depth) is a popular notion of data depth that was introduced and studied by Vardi and Zhang (2000) and Serfling (2002). The *spatial depth* (SPD) of an observation $\mathbf{x} \in \mathbb{R}^d$ with respect to (w.r.t.) a distribution function F on $\mathbb{R}^d$ is defined as $\text{SPD}(\mathbf{x}, F) = 1 - \big\| E_F[u(\mathbf{x} - \mathbf{X})] \big\|$, where $\mathbf{X} \sim F$, and $u(\cdot)$ is the multivariate sign function given by $u(\mathbf{x}) = \|\mathbf{x}\|^{-1}\mathbf{x}$ if $\mathbf{x} \neq \mathbf{0}_d \in \mathbb{R}^d$, and $u(\mathbf{0}_d) = \mathbf{0}_d$. This version of SPD is invariant w.r.t. location shift, orthogonal, and homogeneous scale transformations. SPD is often computed on the standardized version of $\mathbf{X}$ as well. In that case, it is defined as

$$\text{SPD}(\mathbf{x}, F) = 1 - \big\| E_F[u(\mathbf{\Sigma}^{-1/2}(\mathbf{x} - \mathbf{X}))] \big\|,$$

where $\mathbf{\Sigma}$ is a scatter matrix associated with F. One can check that if $\mathbf{\Sigma}$ has the affine equivariance property (see, e.g., Zuo and Serfling, 2000), this version of SPD is affine invariant. To differentiate between these two versions of SPD, we will denote them by SPD$^\circ$ and SPD*, respectively. If $\mathbf{\Sigma} = \lambda \mathbf{I}_d$ for some $\lambda > 0$ (e.g., if F is spherically symmetric, see Fang et al., 1990), then SPD$^\circ$ and SPD* coincide. Throughout this article, the term SPD will be used in a generic sense.

Like other depth functions, SPD provides a center-outward ordering of multivariate data. An observation has higher (respectively, lower) depth if it lies close to (respectively, away from) the center of the distribution. In other words, given an observation $\mathbf{x}$ and a pair of probability distributions F_1 and F_2, if $\text{SPD}(\mathbf{x}, F_1)$ is larger than $\text{SPD}(\mathbf{x}, F_2)$, one would expect $\mathbf{x}$ to come from F_1 instead of F_2. Based on this simple idea, the *maximum depth classifier* was developed by Jornsten (2004); Ghosh and Chaudhuri (2005b). For a J class problem involving distributions $F_1, \ldots, F_J$, the maximum depth classifier based on SPD assigns an observation $\mathbf{x}$ to the j_0-th class, where $j_0 = \text{argmax}_{1 \leq j \leq J} \text{SPD}(\mathbf{x}, F_j)$.



(a) Example **E1**

(b) Example **E2**

Figure 3: $\text{SPD}(\mathbf{x}, F_1)$ and $\text{SPD}(\mathbf{x}, F_2)$ for different values of $\|\mathbf{x}\|$ when $\mathbf{x} \in \mathbb{R}^2$.

In **E1** and **E2**, since the class distributions are spherically symmetric, SPD^* coincides with SPD°, and they become a monotonically decreasing function of the Euclidean norm of $\mathbf{x}$ (see Lemma 7). In Figure 3, we have plotted $\text{SPD}(\mathbf{x}, F_1)$ and $\text{SPD}(\mathbf{x}, F_2)$ for different values of $\|\mathbf{x}\|$ in **E1** and **E2**, where F_1 and F_2 are the distributions of the two classes and $\mathbf{x} \in \mathbb{R}^2$. It is transparent from Figure 3 that the maximum depth classifier based on SPD will fail in both examples. In **E1**, for all values of $\|\mathbf{x}\|$ smaller (respectively, greater) than a constant close to 4, the observations will be classified to the first (respectively, the second) class by the maximum SPD classifier. On the other hand, this classifier will classify all observations to the second class in **E2**. Most of the popular depth functions turn out to be monotonically decreasing functions of the Euclidean norm in the case of a spherically symmetric distribution. So, the maximum depth classifiers based on those depth functions will have similar problems as well.

In Section 2, we develop a modified classifier based on SPD to overcome this limitation of maximum depth classifiers. In the literature, most of the modified depth based classifiers are developed mainly for two class problems (see, e.g., Ghosh and Chaudhuri, 2005b; Dutta and Ghosh, 2012; Li et al., 2012; Lange et al., 2014). For classification problems involving $J(> 2)$ classes, one usually solves $\binom{J}{2}$ binary classification problems taking one pair of classes at a time and then uses either majority voting (see, e.g., Friedman, 1996) or pairwise coupling (see, e.g., Hastie and Tibshirani, 1998) to make the final classification. Unlike those existing methods, our proposed classifier directly addresses the J class problem.

Almost all existing depth based classifiers require ellipticity of class distributions to achieve Bayes optimality. To cope with possible multi-modal as well as non-elliptic population distributions, we construct a localized version of spatial depth (LSPD) in Section 3. In Section 4, we develop a multi-scale classifier based on LSPD. Relevant theoretical results on SPD, LSPD and the resulting classifiers are studied in these sections. In Sections 5 and 6, some simulated and benchmark data sets are analyzed to demonstrate the usefulness of these proposed classifiers. An advantage of SPD over other depth functions is its computational simplicity. Classifiers based on SPD and LSPD can be constructed even when the dimension exceeds the sample size. We deal with such high dimension, low sample size (HDLSS) cases in Section 7, and show that both classifiers turn out to be optimal under a fairly general framework. Several high-dimensional data sets are also analyzed to evaluate their empirical performance. All proofs and mathematical details are given in Appendix A.

2. Bayes Optimality of a Classifier Based on Spatial Depth

Let us assume that $f_1, \ldots, f_J$ are density functions of J elliptically symmetric distributions (Fang et al., 1990) on $\mathbb{R}^d$, where $f_j(\mathbf{x}) = |\mathbf{\Sigma}_j|^{-1/2} g_j(\|\mathbf{\Sigma}_j^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_j)\|)$ for $1 \leq j \leq J$. Here $\boldsymbol{\mu}_j \in \mathbb{R}^d$, $\mathbf{\Sigma}_j$ is a $d \times d$ symmetric and positive definite matrix, and $g_j(\|\mathbf{t}\|)$ is a probability density function of a spherically symmetric distribution on $\mathbb{R}^d$ for $1 \leq j \leq J$. For such classification problems involving general elliptic populations with equal or unequal priors, the next theorem establishes the Bayes optimality of a classifier, which is based on $\mathbf{z}^*(\mathbf{x}) = (z_1^*(\mathbf{x}), \ldots, z_J^*(\mathbf{x}))^T = (\mathrm{SPD}^*(\mathbf{x}, F_1), \ldots, \mathrm{SPD}^*(\mathbf{x}, F_J))^T$.

Theorem 1 *If the densities of J competing classes are elliptically symmetric, the posterior probabilities of these classes satisfy the logistic regression model given by*

$$p(j|\mathbf{x}) = \tilde{p}(j|\mathbf{z}^*(\mathbf{x})) = \frac{\exp(\Phi_j(\mathbf{z}^*(\mathbf{x})))}{[1 + \sum_{k=1}^{(J-1)} \exp(\Phi_k(\mathbf{z}^*(\mathbf{x})))]} \ \textit{for} \ \ 1 \leq j \leq (J-1) \tag{1}$$

$$\textit{and} \ \ p(J|\mathbf{x}) = \tilde{p}(J|\mathbf{z}^*(\mathbf{x})) = \frac{1}{[1 + \sum_{k=1}^{(J-1)} \exp(\Phi_k(\mathbf{z}^*(\mathbf{x})))]}. \tag{2}$$

Here $\Phi_j(\mathbf{z}^*(\mathbf{x})) = \varphi_{j1}(z_1^*(\mathbf{x})) + \cdots + \varphi_{jJ}(z_J^*(\mathbf{x}))$, and φ_{ji}s are appropriate real-valued functions of π_j and f_j for $1 \leq j \leq J$. Consequently, the Bayes rule assigns an observation $\mathbf{x}$ to the class j_0, where $j_0 = \mathrm{argmax}_{1 \leq j \leq J} \ \tilde{p}(j|\mathbf{z}^*(\mathbf{x}))$.

Theorem 1 shows that the Bayes classifier is based on a nonparametric multinomial additive logistic regression model for the posterior probabilities, which is a special case of generalized additive models (GAM) (Hastie and Tibshirani, 1990). If the prior probabilities of J classes are equal, and $f_1, \ldots, f_J$ are all elliptic and unimodal differing only in their locations, this Bayes classifier reduces to the maximum depth classifier (Ghosh and Chaudhuri, 2005b) (see Remark 8 after the proof of Theorem 1 in Appendix A). A special case of Theorem 1 with $\mathbf{\Sigma}_j = \lambda_j \mathbf{I}_d$, where $\lambda_j > 0$ for $1 \leq j \leq J$ is stated below.

Corollary 2 *If the densities of J competing classes are spherically symmetric (i.e., $f_j(\mathbf{x}) = g_j(\|\mathbf{x} - \boldsymbol{\mu}_j\|)$ for $1 \leq j \leq J$), then the posterior probabilities of these classes satisfy the logistic regression model given in Theorem 1 with $\mathbf{z}^*(\mathbf{x})$ replaced by $\mathbf{z}^\circ(\mathbf{x}) = (SPD^\circ(\mathbf{x}, F_1), \ldots, SPD^\circ(\mathbf{x}, F_J))^T$.*

For any fixed i and j, one can calculate the J-dimensional vector $\mathbf{z}^{\circ}(\mathbf{x}_{ji})$ (or, $\mathbf{z}^{*}(\mathbf{x}_{ji})$), where $\mathbf{x}_{ji}$ is the i-th labeled observation from the j-th class for $1 \leq i \leq n_j$ and $1 \leq j \leq J$. These $\mathbf{z}^{\circ}(\mathbf{x}_{ji})$s (or, $\mathbf{z}^{*}(\mathbf{x}_{ji})$s) can be viewed as realizations of the vector of covariates in a nonparametric multinomial additive logistic regression model, where the response corresponds to the class label that belongs to $\{1, \ldots, J\}$. Now, a classifier based on SPD can be constructed by fitting a GAM with the logistic link function. This procedure can be viewed as a multinomial logistic regression in the J-dimensional depth plot. Lange et al. (2014); Li et al. (2012); Mozharovskyi et al. (2015) used such plots for nonparametric classification. Recently, Cuesta-Albertos et al. also considered GAM to construct a depth based classifier for functional data. In practice, we use a random sample $\mathbf{x}_1, \ldots, \mathbf{x}_n$ generated from F to compute the empirical versions of SPD$^{\circ}$ and SPD*, which are given by

$$\mathrm{SPD}^{\circ}(\mathbf{x}, F_n) = 1 - \left\| \frac{1}{n} \sum_{i=1}^{n} u(\mathbf{x} - \mathbf{x}_i) \right\| \quad \text{and} \quad \mathrm{SPD}^{*}(\mathbf{x}, F_n) = 1 - \left\| \frac{1}{n} \sum_{i=1}^{n} u(\widehat{\mathbf{\Sigma}}^{-1/2}(\mathbf{x} - \mathbf{x}_i)) \right\|,$$

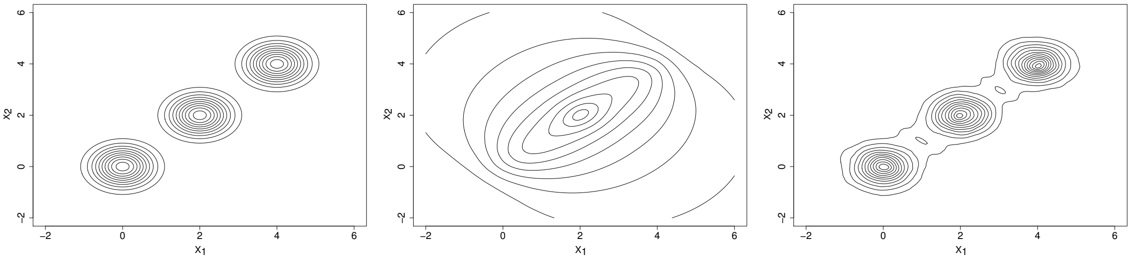
respectively, where $\widehat{\mathbf{\Sigma}}$ is an estimate of $\mathbf{\Sigma}$, and F_n is the empirical distribution of the data $\mathbf{x}_1, \ldots, \mathbf{x}_n$. Clearly, SPD* is affine invariant if $\widehat{\mathbf{\Sigma}}$ has the affine equivariance property. The resulting classifier worked quite well in examples **E1** and **E2**, and we shall see the numerical results later in Section 5.1.

3. Extraction of Small Scale Distributional Features by Localization of Spatial Depth

Under elliptic symmetry, the density function of a class can be expressed as a function of SPD*, and hence the depth contours coincide with the density contours. This is the main mathematical argument used in the proof of Theorem 1. For non-elliptic distributions, where the density function cannot be expressed as a function of SPD, such mathematical arguments are no longer valid. Consider an equal mixture of $\mathrm{N}_d(\mathbf{0}_d, 0.25\mathbf{I}_d)$, $\mathrm{N}_d(2\mathbf{1}_d, 0.25\mathbf{I}_d)$ and $\mathrm{N}_d(4\mathbf{1}_d, 0.25\mathbf{I}_d)$, where $\mathbf{1}_d = (1, \ldots, 1)^T$ denotes a d-dimensional vector with all elements equal to 1. We have plotted the density contours in Figure 4(a) and SPD$^{\circ}$ contours in Figure 4(b) when $d = 2$. In this trimodal distribution, the SPD$^{\circ}$ contours failed to match the density contours. As a second example, we consider a d-dimensional distribution with independent components, where the i-th component is exponential with the scale parameter $d/(d - i + 1)$ for $1 \leq i \leq d$. Figures 5(a) and 5(b) show the density contours and the SPD$^{\circ}$ contours, respectively, when $d = 2$. Even in this example, SPD$^{\circ}$ and density contours differed significantly. We observed a similar picture for contours based on SPD* as well.

To cope with this issue, we suggest a *localization* of SPD. Note that $\mathrm{SPD}^{\circ}(\mathbf{x}, F) = 1 - \|E_F[u(\mathbf{x} - \mathbf{X})]\|$ is constructed by assigning the same weight to each unit vector $u(\mathbf{x} - \mathbf{X})$ and ignoring the significance of the distance between $\mathbf{x}$ and $\mathbf{X}$. By introducing a weight function, which takes account of this distance, one can extract important features related to the local geometry of the data. To capture these local features, we use a kernel function $K(\cdot)$ and define

$$\Gamma_h^{\circ}(\mathbf{x}, F) = E_F[K_h(\mathbf{t})] - \|E_F[K_h(\mathbf{t})u(\mathbf{t})]\|,$$

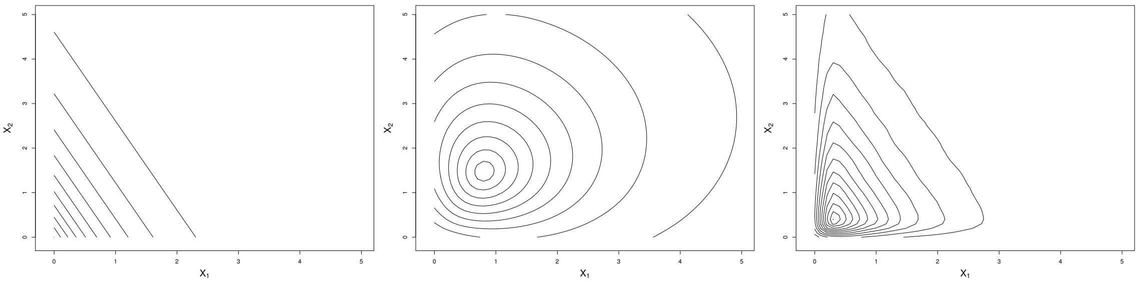


(a) Density

(b) SPD°

(c) $\mathrm{LSPD}^\circ_{h=0.4}$

Figure 4: Contours of density, SPD° and LSPD°_h (with $h = 0.4$) functions for a symmetric, trimodal density function.



(a) Density

(b) SPD°

(c) $\mathrm{LSPD}^\circ_{h=0.25}$

Figure 5: Contours of density, SPD° and LSPD°_h (with $h = 0.25$) functions for the density function $f(x_1, x_2) = 0.5 \exp\{-(x_1 + 0.5x_2)\}I\{x_1 > 0, x_2 > 0\}$.

where $\mathbf{t} = (\mathbf{x} - \mathbf{X})$ and $K_h(\mathbf{t}) = h^{-d}K(\mathbf{t}/h)$. For our theoretical investigation, we will assume K to be a continuous probability density function on $\mathbb{R}^d$ that satisfies the following properties:

(K1) $K(\mathbf{t}) = g_0(\|\mathbf{t}\|)$, where g_0 is a decreasing function with $g_0(0) < \infty$ and $g_0(\|\mathbf{t}\|) \to 0$ as $\|\mathbf{t}\| \to \infty$,

(K2) $K(\mathbf{t})$ has bounded first derivatives, and

(K3) $\int_{\mathbb{R}^d} \|\mathbf{t}\|K(\mathbf{t})d\mathbf{t} < \infty$.

The Gaussian kernel $K(\mathbf{t}) = (\sqrt{2\pi})^{-d} \exp(-\|\mathbf{t}\|^2/2)$ is a possible choice. It is desirable that localized spatial depth (LSPD) approximates the class density, or a monotone function of it for small values of h. This will ensure that the class densities and hence the class posterior probabilities become functions of LSPD as $h \to 0$. On the other hand, one should expect that as $h \to \infty$, LSPD should tend to SPD, or a monotone function of it. However,

$\Gamma^{\circ}_h(\mathbf{x}, F) \to 0$ as $h \to \infty$. So, we re-scale $\Gamma^{\circ}_h(\mathbf{x}, F)$ by an appropriate factor of h to define LSPD° as follows:

$$\text{LSPD}^{\circ}_h(\mathbf{x}, F) = \begin{cases} \Gamma^{\circ}_h(\mathbf{x}, F) & \text{if } h \leq 1, \\ h^d \Gamma^{\circ}_h(\mathbf{x}, F) & \text{if } h > 1. \end{cases} \tag{3}$$

LSPD°_h defined in this way is a continuous function of h. For $d = 2$, Figures 4(c) and 5(c) show that unlike SPD° contours, LSPD°_h contours matched the density contours in both examples. Using $\mathbf{t} = \mathbf{\Sigma}^{-1/2}(\mathbf{x} - \mathbf{X})$ in the definition of $\Gamma^{\circ}_h(\mathbf{x}, F)$, one gets $\Gamma^{*}_h(\mathbf{x}, F)$, and LSPD^{*}_h is defined using $\Gamma^{*}_h(\mathbf{x}, F)$ in the same way. Clearly, LSPD^{*}_h is affine invariant if $\mathbf{\Sigma}$ is affine equivariant. When $\mathbf{\Sigma} = \lambda \mathbf{I}_d$, we obtain $\Gamma^{*}_h(\mathbf{x}, F) = \lambda^{d/2}\Gamma^{\circ}_{h'}(\mathbf{x}, F)$ with $h' = h\sqrt{\lambda}$, and using this expression, one can derive the relation between LSPD^{*}_h and LSPD°_h. The vector $\mathbf{z}^{*}_h(\mathbf{x}) = (\text{LSPD}^{*}_h(\mathbf{x}, F_1), \ldots, \text{LSPD}^{*}_h(\mathbf{x}, F_J))^T$ has the desired behavior as shown in Theorem 3.

Theorem 3 *If $f_1, \ldots, f_J$ are continuous density functions with bounded first derivatives, and $\mathbf{\Sigma}_j$ is the scatter matrix corresponding to the j-th class ($1 \leq j \leq J$), then*
(a) $\mathbf{z}^{*}_h(\mathbf{x}) \to (|\mathbf{\Sigma}_1|^{1/2} f_1(\mathbf{x}), \ldots, |\mathbf{\Sigma}_J|^{1/2} f_J(\mathbf{x}))^T$ as $h \to 0$, and
(b) $\mathbf{z}^{*}_h(\mathbf{x}) \to (K(\mathbf{0})SPD^{*}(\mathbf{x}, F_1), \ldots, K(\mathbf{0})SPD^{*}(\mathbf{x}, F_J))^T$ as $h \to \infty$.

Now, we construct a classifier by plugging in LSPD_h instead of SPD in the GAM framework discussed in equations (1) and (2) of Section 2. Consider the following model for the posterior probabilities:

$$p(j|\mathbf{x}) = \tilde{p}(j|\mathbf{z}^{*}_h(\mathbf{x})) = \frac{\exp(\Phi_j(\mathbf{z}^{*}_h(\mathbf{x})))}{[1 + \sum_{k=1}^{(J-1)} \exp(\Phi_k(\mathbf{z}^{*}_h(\mathbf{x})))]}, \text{ for } 1 \leq j \leq (J-1), \tag{4}$$

$$\text{and } \; p(J|\mathbf{x}) = \tilde{p}(J|\mathbf{z}^{*}_h(\mathbf{x})) = \frac{1}{[1 + \sum_{k=1}^{(J-1)} \exp(\Phi_k(\mathbf{z}^{*}_h(\mathbf{x})))]}. \tag{5}$$

The main implication of part (a) of Theorem 3 is that the classifier constructed using GAM and $\mathbf{z}^{*}_h(\mathbf{x})$ as the covariate tends to the Bayes classifier in a general nonparametric setup as $h \to 0$. On the other hand, part (b) of Theorem 3 implies that for elliptic class distributions, the same classifier tends to the Bayes classifier when $h \to \infty$. When we fit a GAM, the unknown functions Φ_js are estimated nonparametrically. Flexibility of such nonparametric estimates also takes care of the unknown constants $|\mathbf{\Sigma}_j|^{1/2}$ for $1 \leq j \leq J$ and $K(\mathbf{0})$ in the expressions of the limiting values of $\mathbf{z}^{*}_h(\mathbf{x})$ in parts (a) and (b) of Theorem 3, respectively. A special case of Theorem 3 follows by taking $\mathbf{\Sigma}_j = \lambda_j \mathbf{I}_d$ with $\lambda_j > 0$ for all $1 \leq j \leq J$.

Corollary 4 *If $f_1, \ldots, f_J$ are continuous density functions with bounded first derivatives, then*
(a) $\mathbf{z}^{\circ}_h(\mathbf{x}) = (LSPD^{\circ}_h(\mathbf{x}, F_1), \ldots, LSPD^{\circ}_h(\mathbf{x}, F_J))^T \to (f_1(\mathbf{x}), \ldots, f_J(\mathbf{x}))^T$ as $h \to 0$, and
(b) $\mathbf{z}^{\circ}_h(\mathbf{x}) \to (K(\mathbf{0})SPD^{\circ}(\mathbf{x}, F_1), \ldots, K(\mathbf{0})SPD^{\circ}(\mathbf{x}, F_J))^T$ as $h \to \infty$.

If $\mathbf{x}_1, \ldots, \mathbf{x}_n$ is a random sample of size n from F, the empirical version of $\Gamma^{\circ}_h(\mathbf{x}, F)$ is given by

$$\Gamma^{\circ}_h(\mathbf{x}, F_n) = \frac{1}{n}\sum_{i=1}^{n} K_h(\mathbf{t}_i) - \left\| \frac{1}{n}\sum_{i=1}^{n} K_h(\mathbf{t}_i) u(\mathbf{t}_i) \right\|,$$

where $\mathbf{t}_i = (\mathbf{x} - \mathbf{x}_i)$ for $1 \leq i \leq n$. Then $\text{LSPD}^\circ_h(\mathbf{x}, F_n)$ is defined using (3) with $\Gamma^\circ_h(\mathbf{x}, F)$ replaced by $\Gamma^\circ_h(\mathbf{x}, F_n)$. Similarly, we obtain $\Gamma^*_h(\mathbf{x}, F_n)$ and $\text{LSPD}^*_h(\mathbf{x}, F_n)$ by using $\mathbf{t}_i = \widehat{\mathbf{\Sigma}}^{-1/2}(\mathbf{x} - \mathbf{x}_i)$ in the expression stated above. Here $\widehat{\mathbf{\Sigma}}$ is an estimate of $\mathbf{\Sigma}$, and F_n is the empirical distribution of the data $\mathbf{x}_1, \ldots, \mathbf{x}_n$.

We know that $\sup_{\mathbf{x} \in \mathbb{R}^d} |SPD^\circ(\mathbf{x}, F_n) - SPD^\circ(\mathbf{x}, F)|$ goes to 0 almost surely (a.s.) as n goes to infinity (see Gao, 2003). Theorem 5 establishes a similar a.s. uniform convergence of $\text{LSPD}^\circ_h(\mathbf{x}, F_n)$ to its population counterpart $\text{LSPD}^\circ_h(\mathbf{x}, F)$ for a fixed value of h.

Theorem 5 *Assume the density corresponding to the distribution function F to be bounded. Then, for any fixed $h > 0$, $\sup_{\mathbf{x} \in \mathbb{R}^d} |LSPD^\circ_h(\mathbf{x}, F_n) - LSPD^\circ_h(\mathbf{x}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$.*

From the proof of Theorem 5 (see Appendix A), it is easy to check that this a.s. uniform convergence also holds when $h \to \infty$. Under additional moment conditions on F, we obtain this convergence when $h \to 0$ in such a way that $nh^{2d}/\log n \to \infty$ as $n \to \infty$ (see Remarks 9 and 10 after the proof of Theorem 5 in Appendix A).

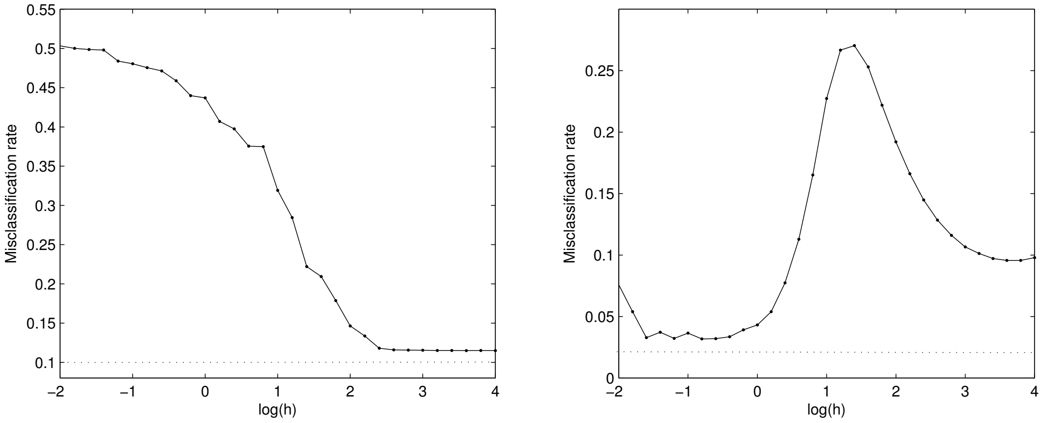
The fact that LSPD tends to a constant multiple of the probability density function as $h \to 0$ is a crucial requirement for limiting Bayes optimality of classifiers based on this local depth function. Agostinelli and Romanazzi (2010) proposed localized versions of simplicial depth and half-space depth, but the relationship between the local depth and the probability density function was established only for $d = 1$. A depth function based on inter-point distances was developed by Lok and Lee (2011) to capture multi-modality in a data set. Chen et al. (2009) defined kernelized spatial depth using a reproducing kernel Hilbert space. Hu et al. (2011) also considered a generalized notion of Mahalanobis depth in reproducing kernel Hilbert spaces. However, there is no result connecting them to the probability density function. In fact, the kernelized spatial depth function becomes degenerate at the value $(1 - 1/\sqrt{2})$ as the tuning parameter goes to zero. Consequently, it becomes non-informative for small values of the tuning parameter. It will be appropriate to note here that none of the preceding authors used their proposed depth functions for constructing classifiers.

Recently, Paindaveine and Van Bever (2013, 2015) proposed a notion of local depth and used it for supervised classification along with other applications. Their version of local depth does not relate to the underlying density function either. At this point, one should note that convergence of local depth function to the underlying density function is an advantageous property for classification. However, this may not always be a desirable property for other applications of data depth (see Paindaveine and Van Bever, 2013, for a detailed discussion).

4. Multi-scale Classification using Localized Spatial Depth

When the class distributions are elliptic, part (b) of Theorem 3 implies that LSPD_h with large values of h will lead to good classifiers. These large values may not be appropriate for non-elliptic class distributions, but part (a) of Theorem 3 implies that LSPD_h with small values of h will lead to good classifiers for general nonparametric models for class densities. However, the empirical version of LSPD_h with small h and the resulting classifier may have their statistical limitations for high-dimensional data.

We now consider two examples to demonstrate the above points. The first example (we call it **E3**) involves two multivariate normal distributions $\mathrm{N}_d(\mathbf{0}_d, \mathbf{I}_d)$ and $\mathrm{N}_d(\mathbf{1}_d, 4\mathbf{I}_d)$. In the second example (we call it **E4**), both the competing classes have trimodal distributions. The first class has the same density as in Figure 4(a) (i.e., an equal mixture of $\mathrm{N}_d(\mathbf{0}_d, 0.25\mathbf{I}_d)$, $\mathrm{N}_d(2\mathbf{1}_d, 0.25\mathbf{I}_d)$ and $\mathrm{N}_d(4\mathbf{1}_d, 0.25\mathbf{I}_d)$), while the second class is an equal mixture of $\mathrm{N}_d(\mathbf{1}_d, 0.25\mathbf{I}_d)$, $\mathrm{N}_d(3\mathbf{1}_d, 0.25\mathbf{I}_d)$ and $\mathrm{N}_d(5\mathbf{1}_d, 0.25\mathbf{I}_d)$. In each of these examples, we considered $d = 5$ and generated a training sample of size 100 from each class. The misclassification rate for the classifier based on LSPD_h° was computed based on a test sample of size 500 (250 observations from each class). This procedure was repeated 100 times to calculate the average misclassification rates for different values of h. Small values of h extracted local distributional features and yielded low misclassification rates in **E4** (see Figure 6(b)). However, those small values of h led to relatively higher misclassification rates in **E3**, while the underlying global elliptic structure was captured well by the proposed classifier for larger values of h (see Figure 6(a)). This provides a strong motivation for adapting a multi-scale approach in constructing the final classifier so that one can harness the strength of different classifiers corresponding to different scales of localization. One would expect that when aggregated judiciously, the multi-scale classifier will lead to an improved misclassification rate. Usefulness of the multi-scale approach in combining different classifiers has been discussed in the classification literature (see, e.g., Kittler et al., 1998; Dzeroski and Zenko, 2004; Ghosh et al., 2005, 2006).



(a) Example **E3**

(b) Example **E4**

Figure 6: Misclassification rates of the Bayes classifier (indicated by dotted lines) and the classifier based on LSPD_h° (indicated by solid curves) in examples **E3** and **E4** for varying choices of h.

A popular way of aggregation is to consider a weighted average of the estimated posterior probabilities computed for different values of h. There are various proposals for the choice of the weight function in the literature. Following Ghosh et al. (2005, 2006), we compute $\widehat{\Delta}_h$, a cross-validation estimate of the misclassification rate of the classifier based on LSPD_h°

(or, LSPD^*_h) and use

$$W(h) \propto \exp\left[-\frac{1}{2}\frac{(\widehat{\Delta}_h - \widehat{\Delta}_0)^2}{\widehat{\Delta}_0(1-\widehat{\Delta}_0)/n}\right]$$

as the weight function, where $\widehat{\Delta}_0 = \min_h \widehat{\Delta}_h$. The exponential function helps to appropriately weigh up (respectively, weigh down) the promising (respectively, the unsatisfactory) classifier resulting from different choices of the smoothing parameter h. We compute $\int W(h)\tilde{g}(h)\tilde{p}(j|\mathbf{z}^*_h(\mathbf{x}))dh$ for the j-th class ($1 \leq j \leq J$), where a probability density function $\tilde{g}$ is used to make the integral finite. Here $\tilde{p}(j|\mathbf{z}^*_h(\mathbf{x}))$ is as defined in equations (4) and (5) of Section 3. If we use very small values of h to classify a test case, then the kernel function used in LSPD_h will put almost zero weights on all observations. Clearly, those small values of h will not be useful for classification. On the other hand, LSPD_h behaves like SPD for large values of h. So, after a certain threshold value, increasing the value of h will not provide any additional information about the distributional features. Therefore, one needs to find suitable lower and upper limits of h to compute the weighted posterior probabilities of different classes. Following Ghosh et al. (2006), we compute the pairwise distances (standardized pairwise distances in the case of LSPD^*_h) among the observations in a class and compute the quantiles of these distances. Let $\lambda_{j,\alpha}$ denote the α-th quantile ($0 < \alpha < 1$) of the pairwise distances for the j-th class with $1 \leq j \leq J$. We use $h_L = \min_j\{\lambda_{j,0.05}\}/3$ as the lower limit of h, and $h_U = 2^r h_L$ as the upper limit of h. Here r is the smallest integer for which we have $\|\mathbf{z}^*_h(\mathbf{x}_{ji}) - \mathbf{z}^*(\mathbf{x}_{ji})\|/\|\mathbf{z}^*(\mathbf{x}_{ji})\| < 0.05$ (or, $\|\mathbf{z}^\circ_h(\mathbf{x}_{ji}) - \mathbf{z}^\circ(\mathbf{x}_{ji})\|/\|\mathbf{z}^\circ(\mathbf{x}_{ji})\| < 0.05$ in case of LSPD°_h) for $1 \leq i \leq n_j$ and $1 \leq j \leq J$. Our final classifier, which we call the LSPD classifier, assigns an observation $\mathbf{x}$ to the class j_0, where

$$j_0 = \underset{1\leq j\leq J}{\text{argmax}} \int_{h_L}^{h_U} W(h)\tilde{g}(h)\tilde{p}(j|\mathbf{z}^*_h(\mathbf{x}))dh.$$

One can choose $\tilde{g}$ to be the uniform distribution on the interval $[h_L, h_U]$. Since we are dealing with a scale parameter h, we take the uniform distribution in the logarithmic scale. In practice, we generate M independent observations $h_1, \ldots, h_M$ from the distribution $\tilde{g}$. For any given $1 \leq j \leq J$ and $\mathbf{x}$, $\int_{h_L}^{h_U} W(h)\tilde{g}(h)\tilde{p}(j|\mathbf{z}^*_h(\mathbf{x}))dh$ is approximated by the average $\sum_{i=1}^{M} W(h_i)\tilde{p}(j|\mathbf{z}^*_{h_i}(\mathbf{x}))/M$.

5. Analysis of Simulated Data Sets

We have analyzed several data sets simulated from elliptic as well as non-elliptic distributions in $\mathbb{R}^5$. In each example, taking an equal number of observations from each of the two competing classes, we generated training and test sets of sizes 200 and 500, respectively. This procedure was repeated 500 times, and the average test set misclassification rates of different classifiers are reported in Tables 1 and 2 along with their corresponding standard errors. To facilitate comparison, the corresponding Bayes risks are reported as well. In all the tables in this article, the best misclassification rate in a data set is indicated by '$*$'. The other figures in bold (if any) are the misclassification rates whose differences from the best misclassification rate were found to be statistically insignificant at the 5% level when the usual large sample test for equality of proportions was used.

For the classifiers based on SPD and LSPD, we wrote our own `R` codes and they are available at the link goo.gl/E5tmd6. Throughout this article, we have used 50 different values of h for multi-scale classification based on LSPD, and the weight function is computed using 5-fold cross-validation method. In this section and in Section 6, we have used SPD* and LSPD$^*_h$ for classification with the usual sample covariance matrix of the j-th class as $\hat{\mathbf{\Sigma}}_j$ for $1 \leq j \leq J$. Any other choice of $\hat{\mathbf{\Sigma}}_j$ has been mentioned at appropriate places.

We compared our proposed classifiers with a pool of classifiers that include parametric classifiers like LDA and QDA, and nonparametric classifiers like those based on k-NN (with the Euclidean metric as the distance function) and KDE (with the Gaussian kernel). For k-NN and KDE, we have used the pooled sample covariance matrix for standardization. Tables 1 and 2 show misclassification rates for the multi-scale versions of k-NN (Ghosh et al., 2005) and KDE (Ghosh et al., 2006) based on the same weight function described in Section 4. For the multi-scale method based on KDE, we have considered 50 equi-spaced values of the bandwidth in the range suggested by Ghosh et al. (2006). For the multi-scale version of k-NN, we considered all possible values of k (see Ghosh et al., 2005, for more details). These multi-scale versions usually had better performance then their single scale analogs with the smoothing parameters chosen by the method of cross-validation.

We also considered support vector machines (SVM) (Hastie et al., 2009) based on the linear kernel (i.e., $K(\mathbf{x},\mathbf{y}) = \langle \mathbf{x},\mathbf{y}\rangle$) and the radial basis function (RBF) kernel (i.e., $K_\gamma(\mathbf{x},\mathbf{y}) = \exp(-\gamma\|\mathbf{x}-\mathbf{y}\|^2)$) to facilitate comparison. We used the codes available at the `R` library `e1071` (Dimitriadou et al., 2011). For the RBF kernel, it has been suggested in the literature to use $\gamma = 1/d$ (see `http://www.csie.ntu.edu.tw/~cjlin/libsvm/`). However, for our numerical work, we considered $\gamma = i/10d$ for $1 \leq i \leq 50$. We also used 25 different values for the box constraint in the interval $[0.1, 100]$, which were equi-spaced in the logarithmic scale. Misclassification rates were computed for these different choices of the tuning parameters, and the best result is reported in the tables for both classifiers.

Misclassification rates are also reported for classification tree (TREE), and a boosted version of TREE known as random forest (RF) (see, e.g., Hastie et al., 2009). For the implementation of TREE and RF, we used the `R` codes available in the libraries `tree` (Ripley, 2011) and `randomForest` (Liaw and Wiener, 2002), respectively. For classification tree, the deviance function was used as a measure of impurity, and the maximum height of the tree was restricted to 31. Nodes with less than 5 observations were never considered for splitting. We have combined the results of 500 trees in RF, where each tree was generated based on 63.2% randomly chosen observations from the training sample. At any stage, only a random subset of $\lfloor\sqrt{d}\rfloor$ out of d variables were considered for splitting. Here $\lfloor t\rfloor$ denotes the largest integer less than, or equal to t.

In addition, we also compared the performance of our classifiers with two depth based classification methods: the classifier based on depth-depth (DD) plot (Li et al., 2012) and the maximum depth classifier based on local depth (LD) (Paindaveine and Van Bever, 2013). The DD classifier fits a polynomial on the depth values corresponding to the two competing classes to construct a separating surface. Three notions of depth were used: Mahalanobis depth, half-space depth and projection depth, where the last two depths were computed based on 500 random projections. For each of these depth functions, we used polynomials of degrees 1, 2 and 3. The best result obtained among all these nine possibilities is reported in Tables 1 and 2. For the maximum LD classifier, we used the `R` library `DepthProc`

(Kosiorowski and Zawadzki, 2016) and considered the best result obtained for different choices of depth and a range of values for the localization parameter. The misclassification rates of the maximum LD classifier was higher than those of the DD classifier in almost all cases, and we do not report those results in this article.

5.1 Examples Involving Elliptic Distributions

Recall examples **E1** and **E2** in Section 2, and example **E3** in Section 4 involving elliptic class distributions. In **E1**, the DD classifier led to the lowest misclassification rate closely followed by SPD and LSPD classifiers (see Table 1), but it did not perform well in **E2**. In this example, SPD and LSPD classifiers significantly outperformed all their competitors. Since the class distributions were elliptic, the SPD classifier had a slight edge over the LSPD classifier in these examples. In view of normality of the class distributions, QDA was expected to have the best performance in **E3**. The DD classifier ranked second here, while SPD and LSPD classifiers performed satisfactorily. In all these examples, the Bayes classifier had non-linear class boundaries. So, LDA and SVM with the linear kernel did not perform well. The performance of SVM with the RBF kernel was relatively better, and it had competitive misclassification rates in **E3**. In all these examples, nonparametric classifiers based on k-NN and KDE yielded much higher misclassification rates compared to SPD and LSPD classifiers.

Table 1: Misclassification rates (in %) of different classifiers in elliptic data sets.

Ex	Bayes	LDA	QDA	SVM	SVM	k-NN	KDE	TREE	RF	DD	SPD	LSPD
	risk			(linear)	(RBF)							
E1	26.50	50.22	51.58	45.46	33.03	39.99	39.16	36.90	31.32	**27.92** $*$	28.32	28.54
		(0.11)	(0.19)	(0.12)	(0.12)	(0.13)	(0.12)	(0.13)	(0.11)	(0.12)	(0.10)	(0.11)
E2	0.00	47.43	42.08	43.92	34.06	36.98	34.29	39.10	34.26	26.68	**8.23** $*$	**8.26**
		(0.11)	(0.12)	(0.11)	(0.12)	(0.13)	(0.15)	(0.13)	(0.11)	(0.13)	(0.11)	(0.10)
E3	10.14	21.56	**11.09** $*$	22.09	11.74	17.86	16.95	19.18	13.77	11.37	11.49	11.64
		(0.09)	(0.07)	(0.09)	(0.07)	(0.09)	(0.08)	(0.13)	(0.08)	(0.08)	(0.07)	(0.07)

5.2 Examples Involving Non-elliptic Distributions

We now consider some examples involving non-elliptic class distributions. Recall the trimodal example **E4** discussed in Section 4. In this example, when the classifiers based on k-NN and KDE were used after standardizing the data set by the pooled sample covariance matrix, they yielded misclassification rates higher than 40%. For KDE, we used a common bandwidth in all directions after standardization. This lead to the use of a large bandwidth in the principal component direction $\frac{1}{\sqrt{d}}\mathbf{1}_d$ (this can be observed from Figure 4(a)). Since the difference between the posterior probabilities of the two classes changes its sign frequently along this direction, use of this large bandwidth makes it difficult to discriminate between the two competing classes. In the k-NN classifier, this standardization leads to the use of a neighborhood which was also elongated along the direction $\frac{1}{\sqrt{d}}\mathbf{1}_d$, and this affected the performance of this classifier. So, we did not standardize the data for these two classifiers, and they outperformed all other classifiers considered here (see Table 2). Classifiers based on SPD* and LSPD* also had poor performance because of this issue with standardization. So we used classifiers based on SPD$^\circ$ and LSPD$^\circ$ in this example. The LSPD$^\circ$ classifier had the third best performance. SVM with the RBF kernel also performed well. All other classifiers had relatively higher misclassification rates. The DD classifier, LDA, QDA and SVM with the linear kernel all misclassified more than 25% of the observations.

The next example (we call it **E5**) is with exponential distributions, where the component variables are independently distributed in both classes. The i-th variable in the first (respectively, the second) class is exponential with scale parameter $d/(d-i+1)$ (respectively, $d/2i$) for $1 \leq i \leq d$. Further, the second class has a location shift such that the difference between the mean vectors of the two classes is $\frac{1}{d}\mathbf{1}_d$. Recall that Figure 5(a) shows the density contours of the first class when $d = 2$. In this example, the RF classifier had the best performance followed by TREE. Here all the measurement variables were independent, and there was significant separation between the two classes in some of the co-ordinate directions. This is one of the main reasons behind the superior performance of both TREE and RF. Classifiers based on DD, SPD* and LSPD* also performed quite well, and their misclassification rates were significantly lower than all other classifiers. The two linear classifiers performed poorly, but QDA had a reasonably good performance in this example. Good performance of QDA was not surprising as the two competing classes are unimodal, while they differ widely in their dispersion structures.

Table 2: Misclassification rates (in %) of different classifiers in non-elliptic data sets.

Ex	Bayes risk	LDA	QDA	SVM (linear)	SVM (RBF)	k-NN	KDE	TREE	RF	DD	SPD	LSPD
E4	2.10	40.45 (0.12)	42.41 (0.11)	36.16 (0.12)	3.28 (0.04)	**2.70** $*$ (0.03)	**2.75** (0.03)	15.52 (0.10)	4.98 (0.07)	30.14 (0.12)	10.07 (0.10)	3.25 (0.04)
E5	2.04	41.17 (0.15)	5.97 (0.05)	32.14 (0.34)	7.12 (0.07)	9.55 (0.08)	9.32 (0.07)	4.82 (0.08)	**2.04** $*$ (0.03)	5.92 (0.05)	5.53 (0.06)	5.42 (0.06)
E6	13.16	49.67 (0.12)	25.77 (0.12)	47.77 (0.15)	29.33 (0.11)	27.44 (0.12)	27.59 (0.11)	38.39 (0.14)	29.73 (0.11)	28.86 (0.14)	**24.15** (0.10)	**24.09** $*$ (0.10)
E7	19.96	50.78 (0.23)	50.48 (0.22)	49.77 (0.07)	46.01 (0.23)	35.29 (0.22)	38.88 (0.24)	34.45 (0.13)	27.62 (0.11)	**26.48** $*$ (0.12)	38.39 (0.20)	40.64 (0.28)

In example **E6**, each class is an equal mixture of four elliptic distributions. The first class constitutes of $N_d(\mathbf{1}_d, S_{0.6}), t_{3,d}(\boldsymbol{\beta}_d, S_{0.7}), N_d(-\mathbf{1}_d, S_{0.8})$ and $t_{3,d}(-\boldsymbol{\beta}_d, S_{0.9})$, while the second class is an equal mixture of $t_{3,d}(\mathbf{1}_d, S_{-0.9}), N_d(\boldsymbol{\beta}_d, 3S_{-0.8}), t_{3,d}(-\mathbf{1}_d, S_{-0.7})$ and $N_d(-\boldsymbol{\beta}_d, 3S_{-0.6})$. Here $t_{3,d}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the d-variate t distribution with 3 degrees of freedom (df), location parameter $\boldsymbol{\mu}$ and scatter matrix $\boldsymbol{\Sigma}$. The vector $\boldsymbol{\beta}_d$ is a d-dimensional vector with the i-th element equal to $(-1)^{i+1}$ for $1 \leq i \leq d$ and the matrix $S_\alpha = ((\alpha^{|i-j|}))_{d\times d}$ for $\alpha \in (-1,1)$ and $1 \leq i, j \leq J$. This example has a complex structure for the class distributions, and both SPD and LSPD classifiers significantly outperformed all their competitors. As the Bayes classifier was far from being linear, LDA and linear SVM did not have satisfactory performance.

Finally, we consider a classification problem between a Cauchy distribution and a skewed Cauchy distribution (Azzalini, 2014) (we call it **E7**). The Cauchy distribution had location parameter $\mathbf{1}_d$ and scatter matrix $0.5\mathbf{I}_d + 0.5\mathbf{1}_d\mathbf{1}_d^T$; while the skewed Cauchy distribution had location parameter $\mathbf{0}_d$, scatter matrix $\mathbf{I}_d$ and asymmetry vector $\mathbf{1}_d$. The DD classifier and RF performed better than other classifiers, but SPD* and LSPD* classifiers yielded relatively higher misclassification rates. Both half-space depth and projection depth used in the DD classifier are robust against outliers generated from heavy-tailed distributions, while the moment based estimates used in both SPD* and LSPD* are non-robust. So, it is better to use robust estimates of $\mathbf{\Sigma}_j$s here. When we used MCD estimates based on 75% of the observations (Rousseeuw and Van Driessen, 1999), the misclassification rates of SPD* and LSPD* classifiers dropped to 31.90% and 32.05%, respectively, with corresponding standard errors of 0.18% and 0.20%.

All these examples clearly demonstrate that the LSPD classifier performs as good as (if not better) popular nonparametric classifiers for non-elliptic, or multi-modal data. This adjustment of the LSPD classifier is automatic in view of the multi-scale approach developed in Section 4.

5.3 Computing Time for SPD and LSPD Classifiers

For a training sample of size n, computation of $\mathbf{z}(\mathbf{x}_{ji})$ for $1 \leq i \leq n_j$ and $1 \leq j \leq J$ requires $O(n^2)$ calculations. Fitting a GAM involves an iterative algorithm, and it is quite difficult to calculate its exact computational complexity. Each iteration requires computations of the order $O(n^2)$ (Wood, 2006). So, the algorithm takes no more than $O(n^2)$ computations to fit a GAM for a finite number of iterations. For the multi-scale classifier based on LSPD, we need to repeat this procedure for M different values of h and then compute the weight function $W(h)$ based on V-fold cross-validation. The overall order of computation remains $O(n^2)$ although the associated constant increases linearly with d, J, M and V. However, one should note that these are offline calculations. Both SPD and LSPD classifiers require $O(n)$ calculations to classify a test case.

Throughout this article, we have used $M = 50$ and $V = 5$ and the `R` library `VGAM` (Yee, 2008) was used to fit GAM. In a single iteration, the average CPU time to determine the weight function $W(h)$ based on cross-validation for the LSPD classifier was 21.83 seconds, while 0.55 seconds were required to fit a GAM using the full training data. The average CPU time to classify the 500 test observations was about 0.01 seconds. All the calculations were done on a desktop computer with an Intel i7 (2.2 GHz) processor having 8 GB RAM.

6. Analysis of Benchmark Data Sets

We have analyzed seven benchmark data sets for further evaluation of our proposed classifiers. The biomedical data set is taken from the CMU data archive (`http://lib.stat.cmu.edu/datasets/`). In this data set, we ignored the observations with missing values. The diabetes data set is available in the `R` library `mclust` (also analyzed in Reaven and Miller, 1979). All other data are taken from the UCI machine learning repository (`http://archive.ics.uci.edu/ml/`). Descriptions of these data sets are available at these sources. Satellite image (satimage) data set has specific training and test samples. For this data set, we report misclassification rates of different classifiers based on this fixed test set. If a classifier had misclassification rate ε, its standard error was computed as $\sqrt{\varepsilon(1-\varepsilon)/(\text{size of the test set})}$. For all other data sets, we formed the training and the test sets by randomly partitioning the data, and this random partitioning was repeated 500 times. Average test set misclassification rates of different classifiers were computed over these 500 partitions, and they are reported in Table 3 along with their corresponding standard errors. Sizes of training and test sets in each partition are also reported in this table. For all classifiers, we used the same tuning procedures as described in Section 5. Codes for the DD classifier are available only for two class problems. In biomedical and Parkinson's data sets, the DD classifier yielded misclassification rates of 12.54% and 14.48%, respectively, with corresponding standard errors of 0.18% and 0.15%. We also used the maximum LD classifier on these real data sets. However, its performance was not satisfactory for most data sets and we do not report those misclassification rates in Table 3.

In biomedical and vehicle data, covariance matrices of the competing classes were different. So, QDA led to significant improvement over LDA, and its misclassification rates were close to the best rate. In both these data sets, the competing classes were nearly elliptic (this can be verified using the diagnostic plots suggested by Li et al., 1997). The SPD classifier utilized this ellipticity of the class distributions to outperform the nonparametric classifiers. The LSPD classifier competed well with the SPD classifier in biomedical data. But, the evidence of ellipticity was much stronger in vehicle data and LSPD had a slightly higher misclassification rate. In diabetes data also, the three competing classes had widely varying covariance structures. As expected, QDA performed better than LDA. Since the class distributions were not elliptic, the SPD classifier yielded a higher misclassification rate than the LSPD classifier, while both TREE and RF outperformed all other classifiers in this data set.

Table 3: Descriptions of the real data sets, and misclassification rates (in %) of different classifiers.

Data set	Biomed	Parkinson's	Diabetes	Wine	Waveform	Vehicle	Satimage
d	4	22	3	13	21	18	36
J	2	2	3	3	3	4	6
Train	100	97	73	100	300	423	4435
Test	94	98	72	78	501	423	2000

Data set	LDA	QDA	SVM (linear)	SVM (RBF)	k-NN	KDE	TREE	RF	SPD	LSPD
Biomed	15.66 (0.14)	**12.57** (0.13)	21.90 (0.13)	**12.76** (0.13)	17.74 (0.15)	16.67 (0.14)	17.69 (0.18)	13.23 (0.14)	**12.53** (0.21)	**12.49 *** (0.15)
Parkinson's	30.93 (0.12)	xxxx xxxx	14.83 (0.12)	13.29 (0.10)	14.42 (0.16)	**11.24 *** (0.12)	16.63 (0.20)	11.68 (0.15)	15.44 (0.15)	14.23 (0.11)
Diabetes	13.86 (0.16)	8.51 (0.13)	10.20 (0.19)	14.93 (0.15)	11.20 (0.13)	11.96 (0.14)	**3.78 *** (0.09)	4.29 (0.10)	9.36 (0.15)	7.93 (0.14)
Wine	2.00 (0.06)	2.46 (0.09)	3.64 (0.09)	1.86 (0.06)	1.98 (0.06)	**1.40 *** (0.05)	10.99 (0.22)	2.12 (0.06)	2.34 (0.08)	1.85 (0.07)
Waveform	19.74 (0.15)	20.78 (0.15)	18.89 (0.07)	16.28 (0.07)	21.23 (0.11)	21.04 (0.11)	28.81 (0.12)	16.45 (0.08)	**15.12 *** (0.06)	15.36 (0.06)
Vehicle	22.49 (0.07)	**16.38** (0.07)	20.59 (0.07)	25.37 (0.08)	21.80 (0.08)	21.21 (0.07)	31.41 (0.10)	25.52 (0.07)	**16.35 *** (0.08)	17.15 (0.08)
Satimage	16.02 (0.82)	14.11 (0.78)	12.95 (0.75)	**8.97** (0.64)	18.00 (0.86)	21.40 (0.92)	18.60 (0.87)	**8.24 *** (0.61)	12.58 (0.74)	12.58 (0.74)

'xxxx': QDA could not be used because of singularity of the estimated class dispersion matrices.

In Parkinson's data, we could not use QDA because of singularity of the estimated class dispersion matrices. So, we used the pooled sample covariance matrix for computation of SPD* and LSPD*. In this data set, all the nonparametric classifiers had significantly lower misclassification rates than LDA, and the classifier based on KDE had the lowest misclassification rate. The performance of the LSPD classifier was also competitive. Since the underlying distributions were non-elliptic, LSPD outperformed the SPD classifier. We observed a similar phenomena in wine data as well. The sample covariance matrices of different classes were nearly singular, and we used the pooled sample covariance matrix for computing SPD* and LSPD*. The classifier based on KDE yielded the lowest misclassification rate, while the LSPD classifier had the second best performance. Although the data dimension was quite high in both data sets, all the competing classes had low intrinsic dimensions (can be estimated using the method described by Levina and Bickel, 2004). So, nonparametric methods like KDE were not affected much by the curse of dimensionality. TREE was the only classifier with a somewhat higher misclassification rate.

In waveform data, the competing class distributions were nearly elliptic and the SPD classifier was expected to perform well. The LSPD classifier is quite flexible, and it yielded a competitive misclassification rate. The class distributions were not normal (can be checked using the method proposed in Royston, 1983) for this data, and did not have low intrinsic dimensions. As a result, LDA, QDA and the nonparametric classifiers had relatively higher misclassification rates.

In satimage data, recall that the results are based on a single training and a single test set. So, the standard errors of the misclassification rates were high for all classifiers, and it is quite difficult to compare the performance of different classifiers. Both RF and SVM with the RBF kernel had lower misclassification rates than other classifiers, while the classifiers based on SPD and LSPD had the next best performance.

7. Classification of High-dimensional Data

A serious practical limitation of many existing depth based classifiers is their computational complexity in high dimensions, and this makes such classifiers impossible to use even for moderately large dimensional data. Besides, depth functions that are based on random simplices formed by the data points (see, e.g., Liu et al., 1999; Zuo and Serfling, 2000) cannot be defined in a meaningful way if the dimension of the data exceeds the sample size. Tukey's half-space depth and projection depth both become degenerate at zero for such high-dimensional data (see, e.g., Dutta et al., 2011). Classification of high-dimensional data presents a substantial challenge to many nonparametric classification tools as well. We have seen in examples **E1** and **E2** (recall Figure 2) that nonparametric classifiers like those based on k-NN and KDE can yield poor performance when the data dimension is large. Some limitations of SVM for classification of high-dimensional data has been noted by Marron et al. (2007); Dutta and Ghosh (2016).

One of our primary motivations behind using SPD is its computational tractability (especially when the dimension is large). If the dimension exceeds the sample size, then the sample covariance matrices become singular, and we cannot use these estimates to define the empirical versions of SPD* and LSPD$^*_h$. So, we use classifiers based on SPD$^\circ$ and LSPD$^\circ_h$. We now assume the following regularity conditions to investigate the behavior of these classifiers for such high-dimensional data.

(C) Consider two independent random vectors $\mathbf{X}_1 = (X_1^{(1)}, \ldots, X_1^{(d)})^T \sim F_j$ and $\mathbf{X}_2 = (X_2^{(1)}, \ldots,\ X_2^{(d)})^T \sim F_i$ for $1 \leq j, i \leq J$.
Further, assume that
(C1) $a_j = \lim_{d\to\infty} d^{-1}\sum_{k=1}^{d} E(X_1^{(k)})^2$ exists, and $d^{-1}\sum_{k=1}^{d}(X_1^{(k)})^2 \overset{a.s.}{\to} a_j$ as $d \to \infty$,
(C2) $b_{ji} = \lim_{d\to\infty} d^{-1}\sum_{k=1}^{d} E(X_1^{(k)}X_2^{(k)})$ exists, and $d^{-1}\sum_{k=1}^{d} X_1^{(k)}X_2^{(k)} \overset{a.s.}{\to} b_{ji}$ as $d \to \infty$.

It is not difficult to verify that for $\mathbf{X}_1 \sim F_j$ $(1 \leq j \leq J)$, if we assume that the sequence of variables $\{X_1^{(k)} - E(X_1^{(k)}) : k = 1, 2, \ldots\}$ centered at their means are independent with uniformly bounded eighth moments (see Theorem 1 (2) in Jung and Marron, 2009, p. 4110), or they are m-dependent processes with some appropriate conditions (see Theorem 2 in de Jong, 1995, p. 350), then the convergence results in (C1) and (C2) hold. Also, if the observations are generated from discrete time ARMA processes, all these conditions are satisfied. Stationarity of such time series is not required here. These assumptions continue to hold if the sequences $\{(X_1^{(k)})^2 - E(X_1^{(k)})^2 : k = 1, 2, \ldots\}$ and $\{X_1^{(k)}X_2^{(k)} - E(X_1^{(k)}X_2^{(k)}) : k = 1, 2, \ldots\}$, where $\mathbf{X}_1 \sim F_j$ and $\mathbf{X}_2 \sim F_i$ for all $1 \leq j, i \leq J$, are *mixingales* satisfying some appropriate conditions (see, e.g., Theorem 2 in de Jong, 1995, p. 350).

Define $\sigma_j^2 = a_j - b_{jj}$ and $\nu_{ji} = b_{jj} - 2b_{ji} + b_{ii}$. For the random vector $\mathbf{X}_1 \sim F_j$, σ_j^2 is the limit of $d^{-1}\sum_{k=1}^{d} Var(X_1^{(k)})$ as $d \to \infty$. If we consider a second independent random vector $\mathbf{X}_2 \sim F_i$ with $i \neq j$, then ν_{ji} is the limit of $d^{-1}\sum_{k=1}^{d}\{E(X_1^{(k)}) - E(X_2^{(k)})\}^2$ as $d \to \infty$. Hall et al. (2005) assumed a similar set of conditions to study the performance of support vector machines (SVM) with the linear kernel and the 1-NN classifier as the data dimension grows to infinity. Similar conditions on observation vectors were also considered by Jung and Marron (2009) to study consistency of principal components of the empirical covariance matrix for high-dimensional data. Under (C1) and (C2), the following theorem describes the behavior of $\mathbf{z}^\circ(\mathbf{x}) = (\text{SPD}^\circ(\mathbf{x}, F_1), \ldots, \text{SPD}^\circ(\mathbf{x}, F_J))^T$ and $\mathbf{z}_h^\circ(\mathbf{x}) = (\text{LSPD}_h^\circ(\mathbf{x}, F_1), \ldots, \text{LSPD}_h^\circ(\mathbf{x}, F_J))^T$ as d grows to infinity.

Theorem 6 *Suppose that the conditions (C1)-(C2) hold, and $\mathbf{X} \sim F_j$ for $1 \leq j \leq J$.*
(a) $\mathbf{z}^\circ(\mathbf{X}) \overset{a.s.}{\to} (c_{j1}, \ldots, c_{jJ})^T = \mathbf{c}_j$ as $d \to \infty$, where $c_{jj} = 1 - \sqrt{\frac{1}{2}}$ and $c_{ji} = 1 - \sqrt{\frac{\sigma_j^2+\nu_{ji}}{\sigma_j^2+\sigma_i^2+\nu_{ji}}}$ for $1 \leq j \neq i \leq J$.
(b) Assume that $h \to \infty$ and $d \to \infty$ in such a way that $\sqrt{d}/h \to 0$ or $A_0(> 0)$. Then, $\mathbf{z}_h^\circ(\mathbf{X}) \overset{a.s.}{\to} g_0(0)\mathbf{c}_j$ or $\mathbf{c}'_j = (g_0(e_{j1}A_0)c_{j1}, \ldots, g_0(e_{jJ}A_0)c_{jJ})^T$ depending on whether $\sqrt{d}/h \to 0$ or A_0, respectively. Here $K(\mathbf{t}) = g_0(\|\mathbf{t}\|)$, $e_{jj} = \sqrt{2}\sigma_j$ and $e_{ji} = \sqrt{\sigma_j^2 + \sigma_i^2 + \nu_{ji}}$ for $j \neq i$.
(c) Assume that $h > 1$, and $\sqrt{d}/h \to \infty$ as $d \to \infty$. Then, $\mathbf{z}_h^\circ(\mathbf{X}) \overset{a.s.}{\to} \mathbf{0}_J$.

The $\mathbf{c}_j$s as well as the $\mathbf{c}'_j$s in the statement of Theorem 6 are *distinct* for all $1 \leq j \leq J$ whenever either $\sigma_j^2 \neq \sigma_i^2$ or $\nu_{ji} \neq 0$ for all $1 \leq j \neq i \leq J$ (see Lemma 11 in Appendix A). In such a case, part (a) of Theorem 6 implies that for large d, $\mathbf{z}^\circ(\mathbf{x})$ becomes degenerate at points depending on the class distributions. So, $\mathbf{z}^\circ(\mathbf{x})$ has good discriminatory power, and our classifier based on SPD$^\circ$ can discriminate well among the J populations. Further, it follows from part (b) that when both d and h grow to infinity in such a way that $\sqrt{d}/h \to 0$ or to a positive constant, $\mathbf{z}_h^\circ(\mathbf{x})$ has good discriminatory power and the classifier based on LSPD$_h^\circ$ can yield low misclassification probability. However, part (c) shows that if $\sqrt{d}$ grows at a rate faster than h, $\mathbf{z}_h^\circ(\mathbf{x})$ converges to the same value $\mathbf{0}_J$ and it becomes non-informative. Consequently, the classifier based on LSPD_h° will lead to a high misclassification probability in this case.

To evaluate the performance of our depth based classifiers for high-dimensional data, we considered examples **E1**-**E7** with $d = 200$. In each example, we generated 20 observations from each class to constitute the training sample, while 250 observations from each class were used to form the test set. We generated 500 training and test sets, and the average test set misclassification rates of the different classifiers along with their corresponding standard errors are reported in Table 4. The Bayes risks were *almost zero* in all these examples, and we have not stated them in Table 4. We did not standardize the data for KDE and k-NN. QDA could not be used in these examples, and we used $\mathbf{I}_d$ instead of the pooled sample covariance matrix for LDA. When the competing classes have equal priors (which is the case in simulated examples), this leads to the Euclidean distance based classifier which classifies an observation to the class having the nearest centroid.

As we have mentioned before, we use SPD° and LSPD° for classification of these high-dimensional data sets. For a single iteration, the LSPD classifier required an average CPU time of 8.82 seconds to compute the weight function $W(h)$, 0.39 seconds for fitting GAM using the full training data, and 0.06 seconds for classification of 500 test cases.

Table 4: Misclassification rates (in %) of different classifiers in simulated data sets.

Example	LDA †	SVM	SVM	k-NN	KDE	TREE	RF	SPD°	LSPD°
		(linear)	(RBF)						
E1	50.93	47.57	28.97	49.71	49.99	45.72	41.95	**0.27** ∗	**0.31**
	(0.13)	(0.09)	(0.38)	(0.06)	(0.06)	(0.15)	(0.14)	(0.03)	(0.03)
E2	45.84	45.69	32.70	49.96	49.92	43.70	39.36	**0.08** ∗	**0.09**
	(0.08)	(0.07)	(0.18)	(0.01)	(0.01)	(0.17)	(0.12)	(0.03)	(0.03)
E3	0.20	0.29	**0.00** ∗	49.99	49.98	27.46	0.28	**0.00** ∗	**0.00** ∗
	(0.01)	(0.01)	(0.00)	(0.01)	(0.01)	(0.12)	(0.17)	(0.00)	(0.00)
E4	34.87	44.28	10.43	**0.19**	**0.20**	38.55	23.57	0.68	**0.13** ∗
	(0.26)	(0.15)	(0.43)	(0.08)	(0.08)	(0.24)	(0.45)	(0.12)	(0.06)
E5	40.83	44.61	13.69	49.98	49.93	18.93	**0.00** ∗	0.84	0.80
	(0.07)	(0.11)	(0.15)	(0.01)	(0.01)	(0.03)	(0.00)	(0.04)	(0.04)
E6	50.11	48.16	31.03	46.52	47.03	48.20	45.00	30.98	**29.76** ∗
	(0.12)	(0.14)	(0.26)	(0.18)	(0.14)	(0.13)	(0.17)	(0.20)	(0.19)
E7	44.74	35.06	31.82	**18.92** ∗	22.91	36.82	22.36	26.33	25.96
	(0.45)	(0.24)	(0.48)	(0.22)	(0.32)	(0.19)	(0.19)	(0.26)	(0.27)

† $\mathbf{I}_d$ was used instead of the pooled sample covariance matrix.

In the first five examples, the two competing classes had separation between either in their locations and/or scales. So, good performance of the SPD° and LSPD° classifiers was expected in view of Theorem 6 and Lemma 11 (see Appendix A). In **E1** and **E2**, recall that the component distributions of the two classes differed only in scales. The SPD° and LSPD° classifiers performed well in these examples, and the former had an edge due to ellipticity of the class distributions. Surprisingly, all other classifiers failed to extract this separability information properly, and had misclassification rates higher than 25%. Since the Bayes class boundaries were highly nonlinear in these two examples, poor performance of linear SVM and LDA was quite expected. Dutta and Ghosh (2016) showed that when one component distribution from the first class and one from the second class differ only in their scales, the k-NN classifier gives a decision in favor of the distribution with a smaller spread (also see Hall et al., 2005). This was the main reason behind the poor performance of the k-NN classifier. Similar arguments can be given for the poor performance of the classifier based on KDE. In these two examples, splitting based on a single variable failed to yield significant reduction in the impurity function (one can see this in Figure 1). So, TREE and RF had relatively higher misclassification rates. In **E3**, the two Gaussian distributions differed in their locations and scales. Barring TREE, k-NN and the classifier based on KDE, all other classifiers yielded misclassification rates close to zero. Since the scale difference between the two classes dominates the location difference, such a poor performance of the classifier based on KDE and k-NN was expected (see the results in Hall et al., 2005; Dutta and Ghosh, 2016). The same explanation holds for **E5** as well. These nonparametric classifiers yielded excellent performance in **E4**, where the component distribution differ only in their locations. However, TREE and RF failed to have satisfactory performance here. Splitting based on linear combinations of the variables may be helpful in **E4** (see Figure 4).

Examples **E6** and **E7** were difficult to deal with. Unlike **E1**-**E5**, none of the classifiers could achieve misclassification rates close to zero in these two examples. Conditions (C1) and (C2) do not hold here, and Theorem 6 is not applicable. The LSPD classifier had the best performance in **E6** (just like the case with $d = 5$ in Section 5). SVM with the RBF kernel and the SPD classifier also led to competitive misclassification rates. Their performance was much better than all other classifiers. In **E7**, the linear classifiers and SVM with the RBF kernel could not perform well. This is also consistent with what we observed in Section 5. Barring TREE, all other classifiers yielded competitive performances in this example. Among them the k-NN classifier led to the lowest misclassification rate.

We also analyzed two high-dimensional benchmark data sets, namely, lightning-2 data and colon data (Alon et al., 1999). The first data set is from the UCR time series classification archive (`http://www.cs.ucr.edu/~eamonn/time_series_data/`), while the other one is taken from the `R` library `rda`. In each case, we formed 500 training and test sets by randomly partitioning each data into two almost equal parts. The average test set misclassification rates of different classifiers are reported in Table 5.

Table 5: Misclassification rates (in %) of different classifiers in real data sets.

Data	d	J	Sample size		LDA [†]	SVM	SVM	k-NN	KDE	TREE	RF	SPD	LSPD
set			Train	Test		(linear)	(RBF)						
Lightning-2	637	2	60	61	31.86 (0.25)	35.64 (0.35)	28.73 (0.32)	29.89 (0.20)	28.11 (0.30)	33.69 (0.34)	**22.08** * (0.34)	27.70 (0.30)	27.16 (0.30)
Colon	2000	2	31	31	**14.47** (0.21)	16.38 (0.23)	21.48 (0.25)	22.47 (0.27)	23.20 (0.28)	28.78 (0.35)	19.28 (0.24)	19.66 (0.31)	19.05 (0.30)

[†] $\mathbf{I}_d$ was used instead of the pooled sample covariance matrix.

Lightning-2 data consist of observations that are realizations of a time series. In this data set, RF had the best performance followed by the LSPD classifier. The SPD classifier also worked well and yielded the third best performance. The class distributions for this data set turn out to be non-elliptic (can be verified using the method proposed by Li et al., 1997) with low intrinsic dimensions (Levina and Bickel, 2004). As a consequence, the classifier based on KDE and k-NN yielded reasonably good performances.

Colon data contain micro-array expression levels of 2000 genes for 'normal' and 'colon cancer' tissues. There was a good linear separation among the observations from the two competing classes, and the linear classifiers lead to low misclassification rates. Among the other classifiers, the LSPD classifier yielded the minimum misclassification rate closely followed by RF and the SPD classifier. These three classifiers were less affected by the curse of dimensionality.

In these high-dimensional benchmark data sets, the data had low intrinsic dimensions due to high correlation among the the measurement variables (Levina and Bickel, 2004). Moreover, data from the competing classes differed mainly in their locations. As a consequence, though the proposed LSPD classifier had a good overall performance, its superiority over the nonparametric methods was not as prominent as it was in the simulated examples.

Acknowledgments

The authors are grateful to Prof. Probal Chaudhuri for his valuable contributions to this manuscript. They are also thankful to the Action Editor and two anonymous reviewers for providing them with several helpful comments. The first author would like to thank Prof. Thomas W. Yee for his help with `VGAM`, and Prof. Jun Li for sharing `R` codes of the DD classifier.

Appendix A. Proofs and Mathematical Details

Lemma 7 *If F has a spherically symmetric density $f(\mathbf{x}) = g(\|\mathbf{x}\|)$ on $\mathbb{R}^d$ with $d > 1$, then $\|E_F[u(\mathbf{x} - \mathbf{X})]\|$ is a non-negative monotonically increasing function of $\|\mathbf{x}\|$.*

Proof of Lemma 7 : In view of spherical symmetry of $f(\mathbf{x})$, $S(\mathbf{x}) = \|E_F[u(\mathbf{x} - \mathbf{X})]\|$ is invariant under orthogonal transformations of $\mathbf{x}$. Consequently, $S(\mathbf{x}) = \eta(\|\mathbf{x}\|)$ for some non-negative function η. Consider now $\mathbf{x}_1$ and $\mathbf{x}_2$ such that $\|\mathbf{x}_1\| < \|\mathbf{x}_2\|$. Using spherical symmetry of $f(\mathbf{x})$, without loss of generality, we can assume $\mathbf{x}_i = (t_i, 0, \dots, 0)^T$ for $i = 1, 2$ such that $|t_1| < |t_2|$. For any $\mathbf{x} = (t, 0, \dots, 0)^T$, we have

$$S(\mathbf{x}) = \left| E_F \left[\frac{(t - X_1)}{\sqrt{(t - X_1)^2 + X_2^2 + \ldots + X_d^2}} \right] \right|,$$

due to spherical symmetry of $f(\mathbf{x})$. For any $\mathbf{x} \in \mathbb{R}^d$ with $d > 1$, $E_F[\|\mathbf{x} - \mathbf{X}\|]$ is a strictly convex function of $\mathbf{x}$ in this case. Consequently, it is a strictly convex function of t. Observe now that $S(\mathbf{x})$ with this choice of $\mathbf{x}$ is the absolute value of the derivative of $E_F[\|\mathbf{x} - \mathbf{X}\|]$ w.r.t. t. This derivative is a symmetric function of t that vanishes at $t = 0$. Hence, $S(\mathbf{x})$ is an increasing function of $|t|$, and this proves that $\eta(\|\mathbf{x}_1\|) < \eta(\|\mathbf{x}_2\|)$. ■

Proof of Theorem 1 : If the population distribution $f_j(\mathbf{x})$ is elliptically symmetric, we have $f_j(\mathbf{x}) = |\mathbf{\Sigma}_j|^{-1/2} g_j(\delta(\mathbf{x}, F_j))$, where $\delta(\mathbf{x}, F_j) = \|\mathbf{\Sigma}_j^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_j)\|$ is the Mahalanobis distance for $1 \leq j \leq J$. Since $\text{SPD}^*(\mathbf{x}, F_j) = 1 - \|E[u(\mathbf{\Sigma}_j^{-1/2}(\mathbf{x} - \boldsymbol{\mu}_j))]\|$ is affine invariant, it is a function of $\delta(\mathbf{x}, F_j)$. Again, as $\mathbf{\Sigma}_j^{-1/2}(\mathbf{X} - \boldsymbol{\mu}_j)$ has a spherically symmetric distribution with its center at the origin, from Lemma 7 it follows that $\mathrm{SPD}^*(\mathbf{x}, F_j)$ is a monotonically decreasing function of $\delta(\mathbf{x}, F_j)$. Therefore, $\delta(\mathbf{x}, F_j)$ is also a function of $\mathrm{SPD}^*(\mathbf{x}, F_j)$ and using this fact $f_j(\mathbf{x})$ can also be expressed as

$$f_j(\mathbf{x}) = \psi_j(\mathrm{SPD}^*(\mathbf{x}, F_j)) \text{ for all } 1 \leq j \leq J,$$

where ψ_j is an appropriate real-valued function that depends on g_j. Now, one can check that

$$\log\Big[\frac{p(j|\mathbf{x})}{p(J|\mathbf{x})}\Big] = \log(\pi_j/\pi_J) + \log\psi_j(\mathrm{SPD}^*(\mathbf{x}, F_j)) - \log\psi_J(\mathrm{SPD}^*(\mathbf{x}, F_J)).$$

for $1 \leq j \leq (J-1)$. Now, if we define $\varphi_{jj}(z) = \log\pi_j + \log\psi_j(z)$ and $\varphi_{ij}(z) = 0$ for $1 \leq j \neq i \leq (J-1)$; and $\varphi_{1J}(z) = \cdots = \varphi_{(J-1)J}(z) = -\log\pi_J - \log\psi_J(z)$, then the proof is complete. ■

Remark 8 *If $f_j(\mathbf{x})$ is unimodal, $\psi_j(z)$ is monotonically increasing for $1 \leq j \leq J$. Moreover, if the distributions differ only in their locations, then the ψ_js are same for all classes. In that case, $f_j(\mathbf{x}) > f_i(\mathbf{x}) \Leftrightarrow \delta(\mathbf{x}, F_j) < \delta(\mathbf{x}, F_i) \Leftrightarrow$ SPD$^*(\mathbf{x}, F_j) >$ SPD$^*(\mathbf{x}, F_i)$ for $1 \leq i \neq j \leq J$, and hence the classifier turns out to be the maximum SPD classifier.*

Proof of Theorem 3(a) : Let $h < 1$. For any fixed $\mathbf{x} \in \mathbb{R}^d$ and the distribution function F_j, we have $\mathrm{LSPD}_h^*(\mathbf{x}, F_j) = E_{F_j}[K_h(\mathbf{t})] - \|E_{F_j}[K_h(\mathbf{t})u(\mathbf{t})]\|$, where $\mathbf{t} = \mathbf{\Sigma}_j{}^{-1/2}(\mathbf{x} - \mathbf{X})$ for $1 \leq j \leq J$. For the first term in the expression of $\mathrm{LSPD}_h^*(\mathbf{x}, F_j)$ above, we have

$$E_{F_j}[K_h(\mathbf{t})] = \int_{\mathbb{R}^d} \frac{1}{h^d} K_h(\mathbf{\Sigma}_j^{-1/2}(\mathbf{x} - \mathbf{v})) f_j(\mathbf{v}) d\mathbf{v} = |\mathbf{\Sigma}_j|^{1/2} \int_{\mathbb{R}^d} K(\mathbf{y}) f_j(\mathbf{x} - h\mathbf{\Sigma}_j^{1/2}\mathbf{y}) d\mathbf{y},$$

where $\mathbf{y} = h^{-1}\mathbf{\Sigma}_j^{-1/2}(\mathbf{x} - \mathbf{v})$. So, using Taylor's expansion of $f_j(\mathbf{x})$, we get

$$E_{F_j}[K_h(\mathbf{t})] = |\mathbf{\Sigma}_j|^{1/2} f_j(\mathbf{x}) - h|\mathbf{\Sigma}_j|^{1/2} \int_{\mathbb{R}^d} K(\mathbf{y})\ (\mathbf{\Sigma}_j^{1/2}\mathbf{y})^T \nabla f_j(\boldsymbol{\xi}) d\mathbf{y},$$

where $\boldsymbol{\xi}$ lies on the line joining $\mathbf{x}$ and $(\mathbf{x} - h\mathbf{\Sigma}_j^{1/2}\mathbf{v})$. Using the Cauchy-Schwartz inequality, one gets $\Big|E_{F_j}[K_h(\mathbf{t})] - |\mathbf{\Sigma}_j|^{1/2} f_j(\mathbf{x})\Big| \leq h|\mathbf{\Sigma}_j|^{1/2}\lambda_j^{1/2} M_j^\circ M_K$, where $M_j^\circ = \sup_{\mathbf{x} \in \mathbb{R}^d} \|\nabla f_j(\mathbf{x})\|$, $M_K = \int \|\mathbf{y}\| K(\mathbf{y}) d\mathbf{y}$, and λ_j is the largest eigenvalue of $\mathbf{\Sigma}_j$. This implies $\Big|E_{F_j}[K_h(\mathbf{t})] - |\mathbf{\Sigma}_j|^{1/2} f_j(\mathbf{x})\Big| \to 0$ as $h \to 0$ for $1 \leq j \leq J$.

For the second term in the expression of $\mathrm{LSPD}_h^*(\mathbf{x}, F_j)$, a similar argument yields

$$\begin{aligned} E_{F_j}[K_h(\mathbf{t})u(\mathbf{t})] &= |\mathbf{\Sigma}_j|^{1/2} \int_{\mathbb{R}^d} K(\mathbf{y}) u(\mathbf{y}) f_j(\mathbf{x} - h\mathbf{\Sigma}_j^{1/2}\mathbf{y}) d\mathbf{y} \\ &= -h|\mathbf{\Sigma}_j|^{1/2} \int_{\mathbb{R}^d} K(\mathbf{y}) u(\mathbf{y})\ (\mathbf{\Sigma}_j^{1/2}\mathbf{y})^T \nabla f_j(\boldsymbol{\xi}) d\mathbf{y} \text{ (as } \int_{\mathbb{R}^d} K(\mathbf{y}) u(\mathbf{y}) d\mathbf{y} = \mathbf{0}). \end{aligned}$$

Now, $\|E_{F_j}[K_h(\mathbf{t})u(\mathbf{t})]\| \leq h|\mathbf{\Sigma}_j|^{1/2}\lambda_j^{1/2} M_j^\circ M_K \to 0$ and this implies that $\mathrm{LSPD}_h^*(\mathbf{x}, F_j) \to |\mathbf{\Sigma}_j|^{1/2} f_j(\mathbf{x})$, as $h \to 0$. Consequently, we have $\mathbf{z}_h^*(\mathbf{x}) \to (|\mathbf{\Sigma}_1|^{1/2} f_1(\mathbf{x}), \ldots, |\mathbf{\Sigma}_J|^{1/2} f_J(\mathbf{x}))^T$ as $h \to 0$. ■

Proof of Theorem 3(b) : Here we consider the case $h > 1$. Take any fixed $\mathbf{x} \in \mathbb{R}^d$ and a j with $1 \leq j \leq J$. For any fixed $\mathbf{t}$, since $K(\mathbf{t}/h) \to K(\mathbf{0})$ as $h \to \infty$ and K is bounded, using Dominated Convergence Theorem (DCT), one can show that $\text{LSPD}^*_h(\mathbf{x}, F_j) \to K(\mathbf{0})\text{SPD}^*(\mathbf{x}, F_j)$ as $h \to \infty$. So, $\mathbf{z}^*_h(\mathbf{x}) \to (K(\mathbf{0})\text{SPD}^*(\mathbf{x}, F_1), \ldots, K(\mathbf{0})$ $\text{SPD}^*(\mathbf{x}, F_J))^T$ as $h \to \infty$. ■

Proof of Theorem 5 : Define the sets $B_n = \{\mathbf{x} = (x_1, \ldots, x_d) : \|\mathbf{x}\| \leq \sqrt{d}n\}$, and $A_n = \{\mathbf{x} :\ n^2 x_i \text{ is an integer and } |x_i| \leq n \text{ for all } 1 \leq i \leq d\}$. Clearly $A_n \subset B_n \subset \mathbb{R}^d$, the set B_n is a closed ball and the set A_n has cardinality $(2n^3+1)^d$. We will prove almost sure (a.s.) uniform convergence on three disjoint sets: (i) A_n, (ii) $B_n \setminus A_n$ and (iii) B_n^c.
Consider any fixed $h \in (0, 1]$. Recall that for this choice of h, $\text{LSPD}^\circ_h(\mathbf{x}, F)$ (see equation (3)) and $\text{LSPD}^\circ_h(\mathbf{x}, F_n)$ are defined as follows:

$$\text{LSPD}^\circ_h(\mathbf{x}, F_n) = \frac{1}{nh^d}\sum_{i=1}^{n} K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) - \Big\|\frac{1}{nh^d}\sum_{i=1}^{n} K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big)u(\mathbf{x}-\mathbf{X}_i)\Big\|, \text{ and}$$

$$\text{LSPD}^\circ_h(\mathbf{x}, F) = \frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)\Big] - \frac{1}{h^d}\Big\|E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)u(\mathbf{x}-\mathbf{X})\Big]\Big\|.$$

(i) Define $\mathbf{Z}_i = K(h^{-1}(\mathbf{x}-\mathbf{X}_i))u(\mathbf{x}-\mathbf{X}_i) - E[K(h^{-1}(\mathbf{x}-\mathbf{X}))u(\mathbf{x}-\mathbf{X})]$ for $1 \leq i \leq n$. Note that $\mathbf{Z}_i$s are independent and identically distributed (i.i.d.) with $E(\mathbf{Z}_i) = \mathbf{0}$ and $\|\mathbf{Z}_i\| \leq 2K(\mathbf{0})$. Fix an $\epsilon > 0$. Using the exponential inequality for sums of i.i.d. random vectors (see Yurinskii, 1976, p. 491), we obtain $P\Big(\|n^{-1}\sum_{i=1}^n \mathbf{Z}_i\| \geq \epsilon\Big) \leq 2e^{-C_0 n\epsilon^2}$. Here C_0 is a positive constant that depends on $K(\mathbf{0})$ and ϵ. This now implies that

$$\begin{aligned}
&P\Big(\Big\|\frac{1}{nh^d}\sum_{i=1}^{n} K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big)u(\mathbf{x}-\mathbf{X}_i)\Big\| - \Big\|\frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)u(\mathbf{x}-\mathbf{X})\Big]\Big\| \geq \epsilon\Big) \\
&\leq P\Big(\Big\|\frac{1}{nh^d}\sum_{i=1}^{n} K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big)u(\mathbf{x}-\mathbf{X}_i) - \frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)u(\mathbf{x}-\mathbf{X})\Big]\Big\| \geq \epsilon\Big) \\
&= P\Big(\Big\|\frac{1}{n}\sum_{i=1}^{n}\mathbf{Z}_i\Big\| \geq h^d\epsilon\Big) \leq 2e^{-C_0 nh^{2d}\epsilon^2}. \qquad (6)
\end{aligned}$$

For a fixed value of h, $\sum_{i=1}^n K(h^{-1}(\mathbf{x}-\mathbf{X}_i))$ is a sum of i.i.d. bounded random variables. Using Bernstein's inequality, we obtain

$$P\Big(\Big|\frac{1}{n}\sum_{i=1}^{n} K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) - E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)\Big]\Big| \geq \epsilon\Big) \leq 2e^{-C_1 n\epsilon^2},$$

for some suitable positive constant C_1. This implies

$$P\Big(\Big|\frac{1}{nh^d}\sum_{i=1}^{n} K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) - \frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)\Big]\Big| \geq \epsilon\Big) \leq 2e^{-C_1 nh^{2d}\epsilon^2}. \qquad (7)$$

Combining (6) and (7), we get $P(|\text{LSPD}^\circ(\mathbf{x}, F_n) - \text{LSPD}^\circ(\mathbf{x}, F)| \geq \epsilon) \leq C_3 e^{-C_4 nh^{2d}\epsilon^2}$ for some suitable constants C_3 and C_4. Since the cardinality of A_n is $(2n^3+1)^d$, we have

$$P\Big(\sup_{\mathbf{x}\in A_n} |\text{LSPD}^\circ(\mathbf{x}, F_n) - \text{LSPD}^\circ(\mathbf{x}, F)| \geq \epsilon\Big) \leq C_3(2n^3+1)^d e^{-C_4 nh^{2d}\epsilon^2}. \qquad (8)$$

Now, $\sum_{n\geq 1}(2n^3+1)^d e^{-C_4 nh^{2d}\epsilon^2} < \infty$. So, an application of Borel-Cantelli lemma implies that $\sup_{\mathbf{x}\in A_n} |\text{LSPD}_h^\circ(\mathbf{x}, F_n) - \text{LSPD}_h^\circ(\mathbf{x}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$.

(ii) Consider the set $B_n \setminus A_n$. Given any $\mathbf{x}$ in $B_n \setminus A_n$, there exists $\mathbf{y} \in A_n$ such that $\|\mathbf{x}-\mathbf{y}\| \leq \sqrt{2}/n^2$. First we will show that $|\text{LSPD}^\circ(\mathbf{y}, F_n) - \text{LSPD}^\circ(\mathbf{x}, F_n)| \overset{a.s.}{\to} 0$ as $n \to \infty$. Using the mean-value theorem, one obtains

$$\left|\frac{1}{nh^d}\sum_{i=1}^n K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) - \frac{1}{nh^d}\sum_{i=1}^n K\Big(\frac{\mathbf{y}-\mathbf{X}_i}{h}\Big)\right| \leq \frac{1}{nh^{d+1}}\sum_{i=1}^n \left|(\mathbf{x}-\mathbf{y})^T \nabla K\Big(\frac{\boldsymbol{\xi}-\mathbf{X}_i}{h}\Big)\right|,$$

where $\boldsymbol{\xi}$ lies on the line joining $\mathbf{x}$ and $\mathbf{y}$. Note that the right hand side is less than $\dfrac{M_K^{'}}{h^{d+1}}\dfrac{\sqrt{2}}{n^2}$, and $M_K^{'} = \sup_{\mathbf{t}} \|\nabla K(\mathbf{t})\|$. This upper bound is free of $\mathbf{x}$, and goes to 0 as $n \to \infty$. Now,

$$\begin{aligned}
&\left\|\frac{1}{nh^d}\sum_{i=1}^n K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big)u(\mathbf{x}-\mathbf{X}_i)\right\| - \left\|\frac{1}{nh^d}\sum_{i=1}^n K\Big(\frac{\mathbf{y}-\mathbf{X}_i}{h}\Big)u(\mathbf{y}-\mathbf{X}_i)\right\| \\
&\leq \left\|\frac{1}{nh^d}\sum_{i=1}^n \Big[K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big)u(\mathbf{x}-\mathbf{X}_i) - K\Big(\frac{\mathbf{y}-\mathbf{X}_i}{h}\Big)u(\mathbf{y}-\mathbf{X}_i)\Big]\right\| \qquad (9) \\
&\leq \left|\frac{1}{nh^d}\sum_{i=1}^n \Big[K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) - K\Big(\frac{\mathbf{y}-\mathbf{X}_i}{h}\Big)\Big]\right| + K(\mathbf{0})\left\|\frac{1}{nh^d}\sum_{i=1}^n [u(\mathbf{x}-\mathbf{X}_i) - u(\mathbf{y}-\mathbf{X}_i)]\right\|.
\end{aligned}$$

We have proved above that the first part converges to 0 in a.s. sense.

For the second part, consider a ball of radius $1/n$ around $\mathbf{x}$ (say, $B(\mathbf{x}, 1/n)$). Now,

$$\begin{aligned}
\left\|\frac{1}{nh^d}\sum_{i=1}^n [u(\mathbf{x}-\mathbf{X}_i) - u(\mathbf{y}-\mathbf{X}_i)]\right\| &\leq \left|\frac{2}{nh^d}\sum_{i=1}^n I[\mathbf{X}_i \in B(\mathbf{x}, 1/n)]\right| + \frac{2n}{h^d}\|\mathbf{x}-\mathbf{y}\| \\
&\leq \frac{2}{h^d}\left|\frac{1}{n}\sum_{i=1}^n I[\mathbf{X}_i \in B(\mathbf{x}, 1/n)] - P[\mathbf{X}_1 \in B(\mathbf{x}, 1/n)]\right| \\
&\quad + \frac{2}{h^d}P[\mathbf{X}_1 \in B(\mathbf{x}, 1/n)] + \frac{2n\sqrt{2}}{n^2 h^d}.
\end{aligned}$$

Note that $I[\mathbf{X}_i \in B(\mathbf{x}, 1/n)]$s are i.i.d. bounded random variables with expectation $P[\mathbf{X} \in B(\mathbf{x}, 1/n)]$. Therefore, a.s. convergence of the first term follows from Bernstein's inequality. Since $P[\mathbf{X} \in B(\mathbf{x}, 1/n)] \leq M_f n^{-d}$ (where $M_f = \sup_{\mathbf{x}} f(\mathbf{x}) < \infty$), the second term converges to 0. For any fixed h, the third term also converges to 0 as $n \to \infty$. So, we have $|\text{LSPD}_h^\circ(\mathbf{x}, F_n) - \text{LSPD}_h^\circ(\mathbf{y}, F_n)| \overset{a.s.}{\to} 0$ as $n \to \infty$.

Similarly, one can prove that $|\text{LSPD}_h^\circ(\mathbf{x}, F) - \text{LSPD}_h^\circ(\mathbf{y}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$. In the arguments above, all the bounds are free from $\mathbf{x}$ and $\mathbf{y}$. We have also proved that $\sup_{\mathbf{y}\in A_n} |\text{LSPD}_h^\circ(\mathbf{y}, F_n) - \text{LSPD}_h^\circ(\mathbf{y}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$. Combining all these results, we have $\sup_{\mathbf{x}\in B_n \setminus A_n} |\text{LSPD}_h^\circ(\mathbf{x}, F_n) - \text{LSPD}_h^\circ(\mathbf{x}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$.

(iii) Now, consider the region outside B_n (i.e., the set B_n^c). First note that

$$\sup_{\mathbf{x}\in B_n^c} |\text{LSPD}_h^\circ(\mathbf{x}, F_n) - \text{LSPD}_h^\circ(\mathbf{x}, F)| \leq \sup_{\mathbf{x}\in B_n^c} \frac{1}{nh^d}\sum_{i=1}^n K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) + \sup_{\mathbf{x}\in B_n^c} \frac{1}{h^d} E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)\Big].$$

We will show that both of these terms become sufficiently small as $n \to \infty$.

Fix an $\epsilon > 0$. We can choose two constants M_1 and M_2 such that $P(\|\mathbf{X}\| \geq M_1) \leq h^d\epsilon/2K(\mathbf{0})$ and $K(\mathbf{t}) \leq h^d\epsilon/2$ when $\|\mathbf{t}\| \geq M_2$. Now, one can check that

$$\frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)\Big] \leq \frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)I(\|\mathbf{X}\| \leq M_1)\Big] + \frac{1}{h^d}K(\mathbf{0})P(\|\mathbf{X}\| > M_1).$$

If $\mathbf{x} \in B_n^c$ and $\|\mathbf{X}\| \leq M_1$, then $h^{-1}\|\mathbf{x}-\mathbf{X}\| \geq h^{-1}|\sqrt{d}n - M_1|$. Choose n large enough so that $|\sqrt{d}n - M_1| \geq M_2 h$, and this implies $K(h^{-1}(\mathbf{x}-\mathbf{X})) \leq h^d\epsilon/2$. So, we obtain

$$\begin{aligned}
\frac{1}{h^d}E\Big[K\Big(\frac{\mathbf{x}-\mathbf{X}}{h}\Big)\Big] &\leq \frac{\epsilon}{2} + \frac{1}{h^d}K(\mathbf{0})P(\|\mathbf{X}\| > M_1) \leq \epsilon, \text{ and}\\
\frac{1}{nh^d}\sum_{i=1}^n K\Big(\frac{\mathbf{x}-\mathbf{X}_i}{h}\Big) &\leq \frac{\epsilon}{2} + \frac{1}{h^d}K(\mathbf{0})\frac{1}{n}\sum_{i=1}^n I(\|\mathbf{X}_i\| > M_1)\\
&\leq \epsilon + \frac{1}{h^d}K(\mathbf{0})\left|\frac{1}{n}\sum_{i=1}^n I(\|\mathbf{X}_i\| > M_1) - P(\|\mathbf{X}\| > M_1)\right|.
\end{aligned}$$

The Glivenko-Cantelli theorem implies that the last term on the right hand side converges to 0 as $n \to \infty$. So, we have $\sup_{\mathbf{x}\in B_n^c} |\mathrm{LSPD}_h^\circ(\mathbf{x}, F_n) - \mathrm{LSPD}_h^\circ(\mathbf{x}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$.

Combining the arguments in parts (i), (ii) and (iii) and for a fixed $h \in (0, 1]$, we get $\sup_{\mathbf{x}} |\mathrm{LSPD}_h^\circ(\mathbf{x}, F_n) - \mathrm{LSPD}_h^\circ(\mathbf{x}, F)| \overset{a.s.}{\to} 0$ as $n \to \infty$. If we have $h > 1$, then this convergence result can be proved in a similar way. For this case, recall that the definition of $\mathrm{LSPD}^\circ(\mathbf{x}, F)$ does not involve the h^d term in the denominator (see equation (3)). ■

Remark 9 *Following the proof of Theorem 5, it is easy to check that a.s. convergence holds when h diverges to infinity with n.*

Remark 10 *The result continues to hold when $h \to 0$ as well. However, for a.s. convergence in part (i) (to use the Borel-Cantelli lemma) we require $nh^{2d}/\log n \to \infty$ as $n \to \infty$. In part (iii), we need M_1 and M_2 to vary with n. Assume the first moment of the density corresponding to F to be finite, and $\int \|\mathbf{t}\|K(\mathbf{t})d\mathbf{t} < \infty$ (which implies that $\|\mathbf{t}\|K(\mathbf{t}) \to 0$ as $\|\mathbf{t}\| \to \infty$). Also, assume that $nh^{2d}/\log n \to \infty$ as $n \to \infty$. We can now choose $M_1 = M_2 = \sqrt{n}$ to ensure that both $P(\|\mathbf{X}\| \geq M_1) \leq h^d\epsilon/2K(\mathbf{0})$ and $K(\mathbf{t}) \leq h^d\epsilon/2$ for $\|\mathbf{t}\| \geq M_2$ hold for a sufficiently large n.*

Proof of Theorem 6(a) : Consider two independent random vectors $\mathbf{X} = (X^{(1)}, \ldots, X^{(d)})^T \sim F_j$ and $\mathbf{X}_1 = (X_1^{(1)}, \ldots, X_1^{(d)})^T \sim F_j$, where $1 \leq j \leq J$. It follows from (C1) and (C2) that $\|\mathbf{X}-\mathbf{X}_1\|/\sqrt{d} \overset{a.s.}{\to} \sqrt{2\sigma_j^2}$ as $d \to \infty$. So, for almost every realization $\mathbf{x}$ of $\mathbf{X} \sim F_j$,

$$\|\mathbf{x}-\mathbf{X}_1\|/\sqrt{d} \overset{a.s.}{\to} \sqrt{2\sigma_i^2} \text{ as } d \to \infty. \tag{10}$$

Next, consider two independent random vectors $\mathbf{X} \sim F_j$ and $\mathbf{X}_1 \sim F_i$ for $1 \leq i \neq j \leq J$. Using (C1) and (C2), we get $\|\mathbf{X}-\mathbf{X}_1\|/\sqrt{d} \overset{a.s.}{\to} \sqrt{\sigma_j^2 + \sigma_i^2 + \nu_{ji}}$ as $d \to \infty$. Consequently, for almost every realization $\mathbf{x}$ of $\mathbf{X} \sim F_j$

$$\|\mathbf{x}-\mathbf{X}_1\|/\sqrt{d} \overset{a.s.}{\to} \sqrt{\sigma_j^2 + \sigma_i^2 + \nu_{ji}} \text{ as } d \to \infty. \tag{11}$$

Let us next consider $\langle \mathbf{x}-\mathbf{X}_1, \mathbf{x}-\mathbf{X}_2\rangle$, where $\mathbf{X} \sim F_j$, $\mathbf{X}_1, \mathbf{X}_2 \sim F_i$ are independent random vectors, and $\langle \cdot, \cdot \rangle$ denotes the inner product in $\mathbb{R}^d$. Therefore, for almost every realization $\mathbf{x}$ of $\mathbf{X}$, arguments similar to those used in (10) and (11) yield

$$\frac{\langle \mathbf{x}-\mathbf{X}_1, \mathbf{x}-\mathbf{X}_2\rangle}{d} \overset{a.s.}{\to} \sigma_j^2 \text{ as } d \to \infty \text{ if } 1 \le i = j \le J, \text{ and} \tag{12}$$

$$\frac{\langle \mathbf{x}-\mathbf{X}_1, \mathbf{x}-\mathbf{X}_2\rangle}{d} \overset{a.s.}{\to} \sigma_j^2 + \nu_{ji} \text{ as } d \to \infty \text{ if } 1 \le i \ne j \le J. \tag{13}$$

Observe now that $\|E_{F_j}[u(\mathbf{x}-\mathbf{X})]\|^2 = \langle E_{F_j}[u(\mathbf{x}-\mathbf{X}_1)], E_{F_j}[u(\mathbf{x}-\mathbf{X}_2)]\rangle = E_{F_j}[\langle u(\mathbf{x}-\mathbf{X}_1), u(\mathbf{x}-\mathbf{X}_2)\rangle]$, where $\mathbf{X}_1, \mathbf{X}_2 \sim F_j$ are independent random vectors for $1 \le j \le J$.

Since we are dealing with expectations of random vectors with bounded norm, a simple application of DCT implies that for almost every realization $\mathbf{x}$ of $\mathbf{X} \sim F_j$ $(1 \le j \le J)$, as $d \to \infty$,

$$\text{SPD}^{\circ}(\mathbf{x}, F_j) \overset{a.s.}{\to} 1 - \sqrt{\frac{1}{2}} \text{ and } \text{SPD}^{\circ}(\mathbf{x}, F_i) \overset{a.s.}{\to} 1 - \sqrt{\frac{\sigma_j^2 + \nu_{ji}}{\sigma_j^2 + \sigma_i^2 + \nu_{ji}}} \text{ for } i \ne j. \tag{14}$$

Thus, for $\mathbf{X} \sim F_j$, we get $z^{\circ}(\mathbf{X}) = (\text{SPD}^{\circ}(\mathbf{X}, F_1), \ldots, \text{SPD}^{\circ}(\mathbf{X}, F_J))^T \overset{a.s.}{\to} \mathbf{c}_j$ as $d \to \infty$. ■

Proof of Theorem 6(b) : Recall that for $h > 1$, $\text{LSPD}_h^{\circ}(\mathbf{x}, F) = E_F[h^d K_h(\mathbf{t})] - \|E_F[h^d K_h(\mathbf{t}) u(\mathbf{t})]\|$. Since we have assumed $\mathbf{X}$s to be standardized, here we get $h^d K_h(\mathbf{t}) = K((\mathbf{x}-\mathbf{X})/h) = g_0(\|\mathbf{x}-\mathbf{X}\|/h)$. Let $\mathbf{X} \sim F_j$ and $\mathbf{X}_i \sim F_i$ with $1 \le i, j \le J$. Using (10) and (11) above, and the continuity of g_0, for almost every realization $\mathbf{x}$ of $\mathbf{X} \sim F_j$, one obtains the following

$$g_0\left(\frac{\|\mathbf{x}-\mathbf{X}_i\|}{\sqrt{d}} \frac{\sqrt{d}}{h}\right) \overset{a.s.}{\to} g_0(0) \text{ or } g_0(e_{ji} A_0),$$

depending on whether $\sqrt{d}/h \to 0$ or A_0. The proof follows from an application of DCT, and the arguments used in the proof of Theorem 6(a). ■

Proof of Theorem 6(c) : Since $g_0(s) \to 0$ as $s \to \infty$, using the same argument as used in the proof of Theorem 6(b), for $\mathbf{X}_i \sim F_i$ and almost every realization $\mathbf{x}$ of $\mathbf{X} \sim F_j$, we have

$$g_0\left(\frac{\|\mathbf{x}-\mathbf{X}_i\|}{\sqrt{d}} \frac{\sqrt{d}}{h}\right) \overset{a.s.}{\to} 0 \text{ as } \sqrt{d}/h \to \infty.$$

The proof now follows from a simple application of DCT. ■

Lemma 11 *Recall $\mathbf{c}_j$ and $\mathbf{c}'_j$ for $1 \le j \le J$ defined in Theorem 6(a) and (b), respectively. For any $1 \le j \ne i \le J$, $\mathbf{c}_j = \mathbf{c}_i$ if and only if $\sigma_j = \sigma_i$ and $\nu_{ji} = \nu_{ij} = 0$. Similarly, $\mathbf{c}'_j = \mathbf{c}'_i$ if and only if $\sigma_j = \sigma_i$ and $\nu_{ji} = \nu_{ij} = 0$.*

Proof of Lemma 11 : The 'if' part is easy to check in both cases. So, it is enough to prove the 'only if' part and that too for the case of $J = 2$. If $\mathbf{c}_1 = (c_{11}, c_{12})^T$ and $\mathbf{c}_2 = (c_{21}, c_{22})^T$ are equal, then we have

$$\frac{\sigma_1^2 + \nu_{12}}{\sigma_1^2 + \sigma_2^2 + \nu_{12}} = 1/2 \text{ and } \frac{\sigma_2^2 + \nu_{12}}{\sigma_1^2 + \sigma_2^2 + \nu_{12}} = 1/2.$$

These two equations hold simultaneously only if $\sigma_1^2 = \sigma_2^2$ and $\nu_{12} = \nu_{21} = 0$.

Consider the case $\mathbf{c}_1' = \mathbf{c}_2'$. Recall that $c_{11}' = g_0(A_0\sqrt{2}\sigma_1)c_{11}$, $c_{22}' = g_0(A_0\sqrt{2}\sigma_2)c_{22}$, $c_{12}' = g_0(A_0\sqrt{\sigma_1^2+\sigma_2^2+\nu_{12}})c_{12}$ and $c_{21}' = g_0(A_0\sqrt{\sigma_2^2+\sigma_1^2+\nu_{21}})c_{21}$. If possible, assume that $\sigma_1 > \sigma_2$. This implies that $A_0\sqrt{\sigma_1^2+\sigma_2^2+\nu_{12}} > A_0\sqrt{2}\sigma_1$ and hence we obtain

$$g_0(A_0\sqrt{2}\sigma_1) > g_0(A_0\sqrt{\sigma_1^2+\sigma_2^2+\nu_{12}}) \quad \text{(since } g_0 \text{ is monotonically decreasing).} \tag{15}$$

Also, if $\sigma_1 > \sigma_2$, we must have

$$1/2 < \frac{\sigma_1^2}{\sigma_1^2+\sigma_2^2} < \frac{\sigma_1^2+\nu_{12}}{\sigma_1^2+\sigma_2^2+\nu_{12}} < 1 \Leftrightarrow 1-\sqrt{1/2} > 1-\sqrt{\frac{\sigma_1^2+\nu_{12}}{\sigma_1^2+\sigma_2^2+\nu_{12}}}. \tag{16}$$

Combining (15) and (16), we have $c_{11}' > c_{21}'$, and this implies $\mathbf{c}_1' \neq \mathbf{c}_2'$. Similarly, if $\sigma_1 < \sigma_2$, we get $c_{12}' > c_{22}'$ and hence $\mathbf{c}_1' \neq \mathbf{c}_2'$. Again, if $\sigma_1 = \sigma_2$ but $\nu_{12} = \nu_{21} > 0$, similar arguments lead to $\mathbf{c}_1' \neq \mathbf{c}_2'$. This completes the proof. ■

References

C. Agostinelli and M. Romanazzi. Local depth. *Journal of Statistical Planning and Inference*, **141**:817–830, 2010.

U. Alon, N. Barkai, D. A. Notterman, K. Gish, D. Mack, and A. J. Leine. Broad pattern of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences, USA*, **96**: 6745–6750, 1999.

A. Azzalini. *The Skew-Normal and Related Families.* 2014. Cambridge University Press, Cambridge.

Y. Chen, X. Dang, H. Peng, and H. L. Bart Jr. Outlier detection with the kernelized spatial depth function. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **31**: 288–305, 2009.

T. M. Cover and P. E. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, **13**:21–27, 1967.

J. A. Cuesta-Albertos, M. Febrero-Bande, and M. Oviedo da la Fuente. The DD^G-classifier in the functional setting. *Test*, forthcoming.

R. M. de Jong. Laws of large numbers for dependent heterogeneous processes. *Econometric Theory*, **11**:347–358, 1995.

E. Dimitriadou, K. Hornik, F. Leisch, D. Meyer, and A. Weingessel. e1071 : Misc functions of the department of statistics (e1071), TU Wien. R package version 1.5-27, 2011. `http://CRAN.R-project.org/package=e1071`.

S. Dutta and A. K. Ghosh. On robust classification using projection depth. *Annals of the Institute of Statistical Mathematics*, **64**:657–676, 2012.

S. Dutta and A. K. Ghosh. On some transformations of high dimension, low sample size data for nearest neighbor classification. *Machine Learning*, **102**:57–83, 2016.

S. Dutta, A. K. Ghosh, and P. Chaudhuri. Some intriguing properties of Tukey's half-space depth. *Bernoulli*, **17**:1420–1434, 2011.

S. Dzeroski and B. Zenko. Is combining classifiers better than selecting the best one? *Machine Learning*, **54**:255–273, 2004.

K. T. Fang, S. Kotz, and K. W. Ng. *Symmetric Multivariate and Related Distributions.* 1990. Chapman & Hall, London.

J. Friedman. Another approach to polychotomous classification. Technical report, Dept. of Statistics, Stanford University, 1996. `http://old.cba.ua.edu/~mhardin/poly.pdf`.

Y. Gao. Data depth based on spatial rank. *Statistics and Probability Letters*, **65**:217–225, 2003.

A. K. Ghosh and P. Chaudhuri. On data depth and distribution free discriminant analysis using separating surfaces. *Bernoulli*, **11**:1–27, 2005a.

A. K. Ghosh and P. Chaudhuri. On maximum depth and related classifiers. *Scandinavian Journal of Statistics*, **32**:328–350, 2005b.

A. K. Ghosh, P. Chaudhuri, and C. A. Murthy. On visualization and aggregation of nearest neighbor classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**:1592–1602, 2005.

A. K. Ghosh, P. Chaudhuri, and D. Sengupta. Classification using kernel density estimates : Multiscale analysis and visualization. *Technometrics*, **48**:120–132, 2006.

P. Hall, J. S. Marron, and A. Neeman. Geometric representation of high dimension low sample size data. *Journal of the Royal Statistical Society: Series B*, **67**:427–444, 2005.

T. Hastie and R. Tibshirani. *Generalized Additive Models.* 1990. Chapman & Hall, London.

T. Hastie and R. Tibshirani. Classification by pairwise coupling. *Annals of Statistics*, **26**: 451–471, 1998.

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* 2009. Springer, New York.

R. Hoberg and K. Mosler. Data analysis and classification with the zonoid depth. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, **72**:45–59, 2006.

Y. Hu, Y. Wang, Y. Wu, Q. Li, and C. Hou. Generalized Mahalanobis depth in the reproducing kernel Hilbert space. *Statistical Papers*, **52**:511–522, 2011.

R. Jornsten. Clustering and classification based on the L_1 data depth. *Journal of Multivariate Analysis*, **90**:67–89, 2004.

S. Jung and J. S. Marron. PCA consistency in high dimension, low sample size context. *Annals of Statistics*, **37**:4104–4130, 2009.

J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **20**:226–239, 1998.

D. Kosiorowski and Z. Zawadzki. *DepthProc: An R Package for Robust Exploration of Multidimensional Economic Phenomena*, 2016.

T. Lange, K. Mosler, and P. Mozharovskyi. Fast nonparametric classification based on data depth. *Statistical Papers*, **55**:49–69, 2014.

E. Levina and P. J. Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in Neural Information Processing Systems (NIPS)*, **17**:777–784, 2004. MIT Press, Cambridge, MA.

J. Li, J. A. Cuesta-Albertos, and R. Liu. Nonparametric classification procedures based on DD-plot. *Journal of the American Statistical Association*, **107**:737–753, 2012.

R. Z. Li, K. T. Fang, and L. X. Zhu. Some Q-Q probabillity plots to test spherical and elliptic symmetry. *Journal of Computational and Graphical Statistics*, **6**:435–450, 1997.

A. Liaw and M. Wiener. Classification and regression by random forest. *R News*, **2**:18–22, 2002.

R. Liu, J. Parelius, and K. Singh. Multivariate analysis of data depth : Descriptive statistics and inference. *Annals of Statistics*, **27**:783–858, 1999.

W. S. Lok and S. M. S. Lee. A new statistical depth function with applications to multimodal data. *Journal of Nonparametric Statistics*, **23**:617–631, 2011.

J. S. Marron, M. J. Todd, and J. Ahn. Distance weighted discrimination. *Journal of the American Statistical Association*, **102**:1267–1271, 2007.

P. Mozharovskyi, K. Mosler, and T. Lange. Classifying real-world data with the DD$^\alpha$ procedure. *Advances in Data Analysis and Classification*, **9**:287–314, 2015.

D. Paindaveine and G. Van Bever. From depth to local depth : a focus on centrality. *Journal of the American Statistical Association*, **105**:1105–1119, 2013.

D. Paindaveine and G. Van Bever. Nonparametrically consistent depth-based classifiers. *Bernoulli*, **21**:62–82, 2015.

G. M. Reaven and R. G. Miller. An attempt to define the nature of chemical diabetes using a multidimensional analysis. *Diabetologia*, **16**:17–24, 1979.

B. Ripley. tree: Classification and regression trees. R package version 1.0-29, 2011.

P. J. Rousseeuw and K. Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, **41**:212–223, 1999.

J. P. Royston. Some techniques for assessing multivariate normality based on the Shapiro-Wilk W. *Technometrics*, **32**:121–133, 1983.

D. W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization.* 2015. Wiley, Hoboken, New Jersey.

R. Serfling. A depth function and a scale curve based on spatial quantiles. In *Statistics and Data Analysis based on L_1-Norm and Related Methods (Y. Dodge ed.)*, pages 25–38. 2002. Birkhaeuser.

Y. Vardi and C. H. Zhang. The multivariate L_1-median and associated data depth. *Proceedings of the National Academy of Sciences, USA*, **97**:1423–1426, 2000.

S. N. Wood. *Generalized Additive Models: An Introduction with R.* 2006. Chapman & Hall/CRC, Boca Raton, FL.

C. Xia, L. Lin, and G. Yang. An extended projection data depth and its applications to discrimination. *Communication in Statistics - Theory and Methods*, **37**:2276–2290, 2008.

Thomas W. Yee. The VGAM package. *R News*, 8(2):28–39, 2008.

V. V. Yurinskii. Exponential inequalities for sums of random vectors. *Journal of Multivariate Analysis*, **6**:473–499, 1976.

Y. Zuo and R. Serfling. General notions of statistical depth function. *Annals of Statistics*, **28**:461–482, 2000.