Kernel Optimization in Discriminant Analysis

Di You, Onur C. Hamsici and Aleix M. Martinez
Dept. Electrical and Computer Engineering
The Ohio State University, Columbus, OH 43210

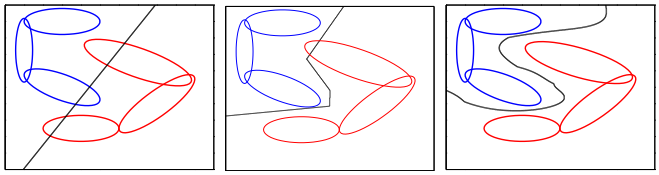
***Abstract*— Kernel mapping is one of the most used approaches to intrinsically derive nonlinear classifiers. The idea is to use a kernel function which maps the original nonlinearly separable problem to a space of intrinsically larger dimensionality where the classes are linearly separable. A major problem in the design of kernel methods is to find the kernel parameters that make the problem linear in the mapped representation. This paper derives the first criterion that specifically aims to find a kernel representation where the Bayes classifier becomes linear. We illustrate how this result can be successfully applied in several kernel discriminant analysis algorithms. Experimental results using a large number of databases and classifiers demonstrate the utility of the proposed approach. The paper also shows (theoretically and experimentally) that a kernel version of Subclass Discriminant Analysis yields the highest recognition rates.**

Index terms: Kernel functions, kernel optimization, feature extraction, discriminant analysis, nonlinear classifiers, face recognition, object recognition, pattern recognition, machine learning.

I. Introduction

Discriminant Analysis (DA) is one of the most popular approaches for feature extraction with broad applications in, for example, computer vision and pattern recognition [10], gene expression analysis [23] and paleontology [22]. The problem with DA algorithms is that each of them makes assumptions on the underlying class distributions. In his ground-breaking work, Fisher [7], [8] derived a DA approach for the two Normally distributed class problem, $N(\mu_1, \mathbf{\Sigma}_1)$ and $N(\mu_2, \mathbf{\Sigma}_2)$, under the assumption of equal covariance matrices, $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2$. Here, μ_i and $\mathbf{\Sigma}_i$ are the mean feature vector and the covariance matrix of the i^{th} class, and $N(.)$ represents the Normal distribution. The assumption of identical covariances (i.e., homoscedasticity) implies that the Bayes (optimal) classifier is linear, which is the reason why we refer to this algorithm as Linear Discriminant Analysis (LDA). LDA thus provides the one-dimensional subspace where the Bayes classification error is smallest in the 2-class homoscedastic problem. A solution for finding the one-dimensional space where the Bayes classification error is minimized in the C-class homoscedastic problem ($\forall\ C \geq 2$) was recently derived in [12].

A major drawback of LDA and of [12] is that they assume the class distributions are homoscedastic, $\mathbf{\Sigma}_i = \mathbf{\Sigma}_j$, $\forall i, j$. This is rarely the case in practise. To resolve this problem, one can first map the original data distributions (with unequal covariances) into a space where these become homoscedastic. This mapping may however result in a space of very large dimensionality. To prevent this, one usually employs the kernel trick [27], [30]. In the kernel trick, the mapping is only intrinsic, yielding a space of the same dimensionality as that of the original representation while still eliminating the nonlinearity of the data by making the class distributions homoscedastic. This is the underlying idea in Kernel DA (KDA) [25], [1] and variants [33], [32], [12]. The need for nonlinear DA is illustrated in Fig. 1(a,c).



(a) (b) (c)

Fig. 1. *Here we show an example of two non-linearly separable class distributions, each consisting of* 3 *subclasses. (a) Classification boundary of LDA. (b) SDA's solution. Note how this solution is piecewise linear (i.e., linear when separating subclasses, but non-linear when classifying classes). (c) KDA's solution.*

The approach described in the preceding paragraph resolves the problem of nonlinearly separable Normal distributions, but still assumes each class can be represented by *a single* Normal distribution. In theory, this can also be learned by the kernel, since multimodality includes nonlinearities in the classifier. In practise however, it makes the problem of finding the appropriate kernel much more challenging. One way to add flexibility to the kernel is to allow for each class to be subdivided into several subclasses. This is the underlying idea behind Subclass DA (SDA) [35]. However, while SDA resolves the problem of multimodally distributed classes, it assumes that these subclass divisions are linearly separable. Note that SDA can actually resolve the problem of nonlinearly separable classes as long as there is a subclass division that results in linearly separable subclasses – yielding a non-linear classifier, Fig. 1(b). The approach will fail when there is no such division. To resolve this problem, we require to derive a subclass-based approach that can deal with nonlinearly separable subclasses [4]. This can be done with the help of a kernel map. In this approach, we need to find a kernel which maps the subclass division into a linearly separable set. We refer to this approach as Kernel SDA (KSDA). Note that KSDA has two unknowns – the number of subclasses and the parameter(s) of the kernel. Hence, finding the appropriate kernel parameters will generally be easier, a point we will formally show in the present paper.

The kernel parameters are the ones that allow us to map a nonlinearly separable problem into a linear one [27]. Surprisingly, to the best of our knowledge, there is not a single method in kernel DA designed to find the kernel parameters which map the problem to a space where the class distributions are linearly separable. To date, the most employed technique is k-fold cross-validation (CV). In CV, one uses a large percentage of the data to train the kernel algorithm. Then, we use the remaining (smaller) percentage of the training samples to test how the classification varies when we use different values in the parameters of the kernel. The parameters yielding the highest recognition rates are

kept. More recently, [31], [16] showed how one can employ the Fisher criterion (i.e., the maximization of the ratio between the kernel between-class scatter matrix and the kernel within-class scatter matrix) to select the kernel parameters. These approaches aim to maximize classification accuracy within the training set. However, neither of them aims to solve the original goal of the kernel map – to find a space where the class distributions (or the samples of different classes) can be separated linearly.

In this paper, we derived an approach whose goal is to specifically map the original class (or subclass) distributions into a kernel space where these are best separated by a hyperplane (wrt Bayes). The proposed approach also aims to maximize the distance between the distributions of different classes, thus maximizing generalization. We apply the derived approach to three kernel versions of DA, namely LDA, Nonparametric DA (NDA) and SDA. We show that the proposed techniques generally achieves higher classification accuracies than the CV and Fisher criteria defined in the preceding paragraph. Before we present the derivations of our criterion, we introduce a general formulation of DA common to most variants. We also derived kernel versions for NDA and SDA.

II. The Metrics of Discriminant Analysis

DA is a supervised technique for feature extraction and classification. Theoretically, its advantage over unsupervised techniques is given by it providing that representation where the underlying class distributions are best separated. Unfortunately, due to the number of possible solutions, this goal is not always fulfilled in practice [23]. With infinite time or computational power, one could always find the optimal representation. With finite time and resources, it is generally impossible to account for all the possible linear combinations of features, let alone a set of nonlinear combinations. This means that one needs to define criteria that can find an appropriate solution under some general, realistic assumptions.

The least-squares extension of Fisher's criterion [8], [10] is arguably the most known. In this solution, LDA employs two symmetric, positive semi-definite matrices, each defining a metric [23]. One of these metrics should measure within-class differences and, as such, should be minimized. The other metric should account for between-class dissimilarity and should thus be maximized. Classical choices for the first metric are the within-class scatter matrix $\mathbf{S}_W$ and the sample covariance matrix $\mathbf{\Sigma}_X$, while the second metric is usually given by the between-class scatter matrix $\mathbf{S}_B$. The sample covariance matrix is defined as $\mathbf{\Sigma}_X = n^{-1}\sum_{i=1}^{n}(\mathbf{x}_i-\mu)(\mathbf{x}_i-\mu)^T$, where $\mathbf{X}=\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ are the n training samples, $\mathbf{x}_i \in \mathbb{R}^p$, and $\mu = n^{-1}\sum_{i=1}^{n}\mathbf{x}_i$ is the sample mean. The between-class scatter matrix is given by $\mathbf{S}_B = \sum_{i=1}^{C} p_i (\mu_i-\mu)(\mu_i-\mu)^T$, where $\mu_i = n_i^{-1}\sum_{j=1}^{n_i}\mathbf{x}_{ij}$ is the sample mean of class i, $\mathbf{x}_{ij}$ is the j^{th} sample of class i, n_i is the number of samples in that class, C is the number of classes, and $p_i = n_i/n$ is the prior of class i. LDA's solution is then given by the generalized eigenvalue decomposition equation $\mathbf{\Sigma}_X^{-1}\mathbf{S}_B\mathbf{V} = \mathbf{V}\Lambda$, where the columns of $\mathbf{V}$ are the eigenvectors, and Λ is a diagonal matrix of corresponding eigenvalues.

To loosen the parametric restriction on the above defined metrics, Fukunaga and Mantock defined NDA [11], where the between-class scatter matrix is changed to a non-parametric version, $\mathbf{S}_b = \sum_{i=1}^{C}\sum_{\substack{j=1\\ j\neq i}}^{C}\sum_{l=1}^{n_i}\alpha_{ijl}(\mathbf{x}_{il}-\mu_{il}^{j})(\mathbf{x}_{il}-\mu_{il}^{j})^T$, where μ_{il}^{j} is the sample mean of the k-nearest samples to the samples $\mathbf{x}_{il}$ that do not belong to class i, and α_{ijl} is a scale factor that deemphasizes large values (i.e. outliers). Alternatively, Friedman [9] proposed to add a regularizing parameter to the within-class measure, allowing for the minimization of the generalization error. This regularizing parameter can be learned using CV, yielding the method Regularized DA (RDA). Another variant of LDA is given by Loog et al. [20], who introduced a weighted version of the metrics in an attempt to downplay the roles of the class distributions that are farthest apart. More formally, they noted that the above introduced Fisher criterion for LDA can be written as $\sum_{i=1}^{C-1}\sum_{j=i+1}^{C} p_i p_j \beta_{ij} tr\left(\left(\mathbf{V}^T\mathbf{S}_W\mathbf{V}\right)^{-1}\left(\mathbf{V}^T\mathbf{\Sigma}_{ij}\mathbf{V}\right)\right)$, where $\mathbf{\Sigma}_{ij} = (\mu_i-\mu_j)(\mu_i-\mu_j)^T$, and β_{ij} are the weights. In Fisher's LDA, all $\beta_{ij} = 1$. Loog et al. suggest to make these weights inverse proportional to their pairwise accuracy (defined as one minus the Bayes error). Similarly, we can define a weighted version of the within-class scatter matrix $\mathcal{S}_W = \sum_{c=1}^{C}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\gamma_{ckl}(\mathbf{x}_{ck}-\mathbf{x}_{cl})(\mathbf{x}_{ck}-\mathbf{x}_{cl})^T$. In LDA, γ_{ckl} are all equal to one. In its weighted version, γ_{ckl} are defined according to the importance of each sample in classification. Using the same notation, we can also define a nonparametric between-class scatter matrix as $\mathcal{S}_B = \sum_{i=1}^{C-1}\sum_{j=1}^{n_i}\sum_{k=i+1}^{C}\sum_{l=1}^{n_k}\rho_{ijkl}(\mathbf{x}_{ij}-\mathbf{x}_{kl})(\mathbf{x}_{ij}-\mathbf{x}_{kl})^T$, where ρ_{ijkl} are the weights. Note that in these two definitions, the priors have been combined with the weights to provide a more compact formulation.

All the methods introduced in the preceding paragraphs assume the class distributions are unimodal Gaussians. To address this limitation, Subclass DA (SDA) [35] defines a multimodal between-*sub*class scatter matrix,

$$\mathbf{\Sigma}_B = \sum_{i=1}^{C-1}\sum_{j=1}^{H_i}\sum_{k=i+1}^{C}\sum_{l=1}^{H_k} p_{ij}p_{kl}(\mu_{ij}-\mu_{kl})(\mu_{ij}-\mu_{kl})^T, \quad (1)$$

where $p_{ij} = n_{ij}/n$ is the prior of the j^{th} subclass of class i, n_{ij} is the number of samples in the j^{th} subclass of class i, H_i is the number of subclasses in class i, $\mu_{ij} = \frac{1}{n_{ij}}\sum_{k=1}^{n_{ij}}\mathbf{x}_{ijk}$ is the sample mean of the j^{th} subclass in class i, and $\mathbf{x}_{ijk}$ denotes the k^{th} sample in the j^{th} subclass in class i.

The algorithms summarized thus far assume the class (or subclass) distributions are homoscedastic. To deal with heteroscedastic (i.e., non-homoscedastic) distributions, [19] defines a within-class similarity metric using the Chernoff distance, yielding an algorithm we will refer to as Heteroscedastic LDA (HLDA). Alternatively, one can use an embedding approach such as Locality Preserving Projection (LPP) [15]. LPP finds that subspace where the structure of the data is locally preserved, allowing for nonlinear classifications. An alternative to these algorithms is to employ a kernel function which intrinsically maps the original data distributions to a space where these adapt to the assumptions of the approach in use. KDA [25], [1] redefines the within- and between-class scatter matrices in the kernel space to derive feature extraction algorithms that are nonlinear in the original space but linear in the kernel one. This is achieved by means of a mapping function $\phi(.) : \mathbb{R}^p \to \mathcal{F}$. The sample covariance and between-class scatter matrices in the kernel space are given by $\mathbf{\Sigma}_X^{\Phi} = n^{-1}\sum_{i=1}^{n}(\phi(\mathbf{x}_i)-\mu^{\phi})(\phi(\mathbf{x}_i)-\mu^{\phi})^T$ and $\mathbf{S}_B^{\Phi} = \sum_{i=1}^{C} p_i\left(\mu_i^{\phi}-\mu^{\phi}\right)\left(\mu_i^{\phi}-\mu^{\phi}\right)^T$, where $\mu^{\phi} = \frac{1}{n}\sum_{i=1}^{n}\phi(\mathbf{x}_i)$ is

the kernel sample mean, and $\mu_i^\phi = \frac{1}{n_i}\sum_{j=1}^{n_i}\phi(\mathbf{x}_{ij})$ is the kernel sample mean of class i.

Unfortunately, the dimensionality of $\mathcal{F}$ may be too large. To bypass this problem, one generally uses the kernel trick, which works as follows. Let $\mathbf{A}^\Phi$ and $\mathbf{B}^\Phi$ be two metrics in the kernel space and $\mathbf{V}^\Phi$ the projection matrix obtained by $\mathbf{A}^\Phi\mathbf{V}^\Phi = \mathbf{B}^\Phi\mathbf{V}^\Phi\Lambda^\Phi$. We know from the Representer's Theorem [30] that the resulting projection matrix can be defined as a linear combination of the samples in the *kernel space* $\Phi(\mathbf{X})$ with the coefficient matrix Γ, i.e., $\mathbf{V}^\Phi = \Phi(\mathbf{X})\Gamma$. Hence, to calculate the projection matrix, we need to obtain the coefficient matrix by solving $\mathbf{A}\Gamma = \mathbf{B}\Gamma\Lambda^\Phi$, where $\mathbf{A} = \Phi(\mathbf{X})^T\mathbf{A}^\Phi\Phi(\mathbf{X})$ and $\mathbf{B} = \Phi(\mathbf{X})^T\mathbf{B}^\Phi\Phi(\mathbf{X})$ are the two metrics that need to be maximized and minimized. Using this trick, the metric for $\mathbf{\Sigma}_X^\Phi$ is given by $\mathbf{B}_{\mathbf{\Sigma}_X^\Phi} = \Phi(\mathbf{X})^T\mathbf{\Sigma}_X^\Phi\Phi(\mathbf{X}) = n^{-1}\sum_{i=1}^{n}\Phi(\mathbf{X})^T(\phi(\mathbf{x}_i)-\mu^\phi)(\phi(\mathbf{x}_i)-\mu^\phi)^T\Phi(\mathbf{X}) = \frac{1}{n}\mathbf{K}(\mathbf{I}-\mathbf{P}_n)\mathbf{K}$, where $\mathbf{K} = \Phi(\mathbf{X})^T\Phi(\mathbf{X})$ is the kernel (Gram) matrix and $\mathbf{P}_n$ is the $n\times n$ matrix with each of its element equal to $1/n$.

Similarly, $\mathbf{B}_{\mathbf{S}_W^\Phi} = \frac{1}{C}\sum_{i=1}^{C}\Phi(\mathbf{X})^T\mathbf{\Sigma}_i^\Phi\Phi(\mathbf{X}) = \frac{1}{C}\sum_{i=1}^{C}\frac{1}{n_i}\mathbf{K}_i(\mathbf{I}-\mathbf{P}_{n_i})\mathbf{K}_i^T$, where $\mathbf{\Sigma}_i^\Phi = \frac{1}{n_i}\sum_{j=1}^{n_i}(\phi(\mathbf{x}_{ij})-\mu_i^\phi)(\phi(\mathbf{x}_{ij})-\mu_i^\phi)^T$ is the kernel within-class covariance matrix of class i, and $\mathbf{K}_i = \Phi(\mathbf{X})^T\Phi(\mathbf{X}_i)$ is the subset of the kernel matrix for the samples in class i. The metric for $\mathbf{S}_B^\Phi$ can be obtained as $\mathbf{A}_{\mathbf{S}_B^\Phi} = \sum_{i=1}^{C}p_i(\mathbf{K}_i\mathbf{1}_{n_i}-\mathbf{K}\mathbf{1}_n)(\mathbf{K}_i\mathbf{1}_{n_i}-\mathbf{K}\mathbf{1}_n)^T$, where $\mathbf{1}_{n_i}$ is a vector with all elements equal to $1/n_i$. The coefficient matrix for KDA is given by $\mathbf{B}_{KDA}^{-1}\mathbf{A}_{KDA}\Gamma_{KDA} = \Gamma_{KDA}\Lambda_{KDA}$, where $\mathbf{B}_{KDA}$ can be either $\mathbf{B}_{\mathbf{\Sigma}_X^\Phi}$ or $\mathbf{B}_{\mathbf{S}_W^\Phi}$ and $\mathbf{A}_{KDA} = \mathbf{A}_{\mathbf{S}_B^\Phi}$.

We can similarly derive kernel approaches for the other methods introduced above. For example, in Kernel NDA (KNDA), the metric $\mathbf{A}$ is obtained by defining its corresponding scatter matrix in the kernel space as

$$\begin{aligned}\mathbf{A}_{KNDA} &= \Phi(\mathbf{X})^T\mathbf{S}_b^\Phi\Phi(\mathbf{X})\\ &= \sum_{i=1}^{C}\sum_{\substack{j=1\\ j\neq i}}^{C}\sum_{l=1}^{n_i}\alpha_{ijl}^\phi(\mathbf{k}_{il}-\mathbf{M}_{il}^j\mathbf{1}_k)(\mathbf{k}_{il}-\mathbf{M}_{il}^j\mathbf{1}_k)^T,\end{aligned}$$

where $\mathbf{k}_{il} = \Phi(\mathbf{X})^T\phi(\mathbf{x}_{il})$ is the kernel space representation of the sample $\mathbf{x}_{il}$, $\mathbf{M}_{il}^j = \Phi(\mathbf{X})^T\Phi(\mathbf{X}_{il}^j)$ is the kernel matrix of the k-nearest neighbors of $\mathbf{x}_{il}$, $\mathbf{X}_{il}^j$ is a matrix whose columns are the k-nearest neighbors of $\mathbf{x}_{il}$, and α_{ijl}^ϕ is the normalizing factor computed in the kernel space.

Kernel SDA (KSDA) maximizes the kernel between-subclass scatter matrix $\mathbf{\Sigma}_B^\Phi$ [4]. This matrix is given by replacing the subclass means of (1) with the kernel subclass means $\mu_{ij}^\phi = n_{ij}^{-1}\sum_{k=1}^{n_{ij}}\phi(\mathbf{x}_{ijk})$. Now, we can use the kernel trick to obtain the matrix to be maximized, $\mathbf{A}_{KSDA} =$

$$\sum_{i=1}^{C-1}\sum_{j=1}^{H_i}\sum_{k=i+1}^{C}\sum_{l=1}^{H_k}p_{ij}p_{kl}(\mathbf{K}_{ij}\mathbf{1}_{ij}-\mathbf{K}_{kl}\mathbf{1}_{kl})(\mathbf{K}_{ij}\mathbf{1}_{ij}-\mathbf{K}_{kl}\mathbf{1}_{kl})^T,$$

where $\mathbf{K}_{ij} = \Phi(\mathbf{X})^T\Phi(\mathbf{X}_{ij})$ is the kernel matrix of the samples in the j^{th} subclass of class i, and $\mathbf{1}_{ij}$ is a $n_{ij}\times 1$ vector with all elements equal to $1/n_{ij}$.

If we are to successfully employ the above derived approaches in practical settings, it is imperative that we define criteria to optimize these parameters. The classical approach to determine the parameters of the kernel is CV, where we divide the training data into k parts: $(k-1)$ of them for training the algorithm with distinct values for the parameters of the kernel, and the remaining one for validating which of these values results in higher (average) classification rates. This solution has three major drawbacks. First, the kernel parameters are only optimized for the training data, not the distributions [36]. Second, CV is computationally expensive and may become very demanding for large data-sets. Third, not all the training data can be used to optimize the parameters of the kernel. To avoid these problems, [31] defines a criterion to maximize the kernel between-class difference and minimize the kernel within-class scatter – as Fisher had originally proposed but now applied to the selection of the kernel parameters. This method was shown to yield higher classification accuracies than CV in a variety of problems. A related approach [16] is to redefine the kernelized Fisher criterion as a convex optimization problem. Alternatively, Ye et al. [34] have proposed a kernel version of RDA where the kernel is learned as a linear combination of a set of pre-specified kernels. However, these approaches do not guarantee that the kernel or kernel parameters we choose will result in homoscedastic distributions in the kernel space. This would be ideal, because it would guarantee that the Bayes classifier (which is the one with the smallest error in that space) is linear.

The main contribution of this paper is to derive a criterion to find a kernel which maps the original class distributions to homoscedastic ones while keeping them as far apart from each other as possible. This criterion is related to the approach presented in [13] where the goal was to optimize a distinct version of homoscedasticity defined in the complex sphere. The criterion we derive in this paper could be extended to work in the complex sphere and is thus a more general approach.

III. Maximizing Homoscedasticity

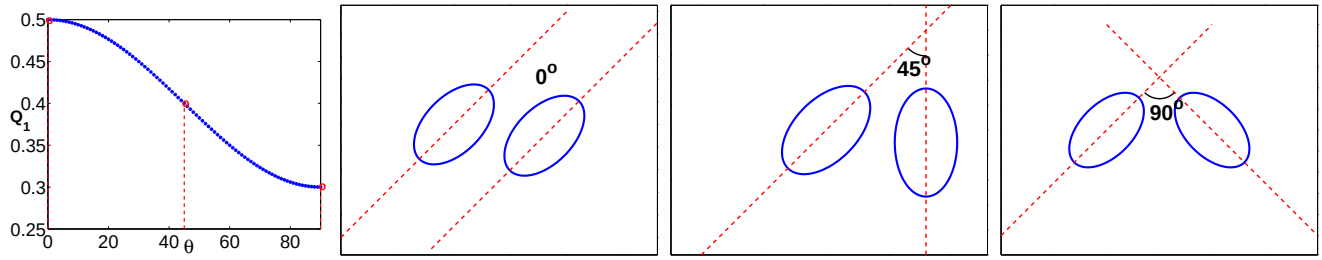
To derive our homoscedastic criterion, we need to answer the following question. What is a good measure of homoscedasticity? That is, we need to define a criterion which is maximized when all class covariances are identical. The value of the criterion should also decrease as the distributions become more different. We now present a key result applicable to this end.

Theorem 1: Let $\mathbf{\Sigma}_i^\Phi$ and $\mathbf{\Sigma}_j^\Phi$ be the kernel covariance matrices of two Normal distributions in the kernel space defined by the function $\phi(.)$. Then, $Q_1 = \frac{tr(\mathbf{\Sigma}_i^\Phi\mathbf{\Sigma}_j^\Phi)}{tr(\mathbf{\Sigma}_i^{\Phi^2})+tr(\mathbf{\Sigma}_j^{\Phi^2})}$ takes the maximum value of .5 when $\mathbf{\Sigma}_i^\Phi = \mathbf{\Sigma}_j^\Phi$, i.e., when the two Normal distributions are homoscedastic in the kernel space.

Proof: $\mathbf{\Sigma}_i^\Phi$ and $\mathbf{\Sigma}_j^\Phi$ are two $p\times p$ positive semi-definite matrices with spectral decompositions $\mathbf{\Sigma}_i^\Phi = \mathbf{V}_i^\Phi\Lambda_i^\Phi\mathbf{V}_i^{\Phi^T}$, where $\mathbf{V}_i^\Phi = \left(\mathbf{v}_{i_1}^\phi,\ldots,\mathbf{v}_{i_p}^\phi\right)$ and $\Lambda_i^\Phi = diag\left(\lambda_{i_1}^\phi,\ldots,\lambda_{i_p}^\phi\right)$ are the eigenvector and eigenvalue matrices.

The denominator of Q_1, $tr(\mathbf{\Sigma}_i^{\Phi^2})+tr(\mathbf{\Sigma}_j^{\Phi^2})$, only depends on the selection of the kernel. For a fixed kernel (and fixed kernel parameters), its value is constant regardless of any divergence between $\mathbf{\Sigma}_i^\Phi$ and $\mathbf{\Sigma}_j^\Phi$. Hence, $tr(\mathbf{\Sigma}_i^{\Phi^2})+tr(\mathbf{\Sigma}_j^{\Phi^2}) = tr(\Lambda_i^{\Phi^2})+tr(\Lambda_j^{\Phi^2})$. We also know that $tr(\mathbf{\Sigma}_i^\Phi\mathbf{\Sigma}_j^\Phi)\leq tr(\Lambda_i^\Phi\Lambda_j^\Phi)$, with the equality holding when $\mathbf{V}_i^{\Phi^T}\mathbf{V}_j^\Phi = \mathbf{I}$ [28], i.e., the eigenvectors of $\mathbf{\Sigma}_i^\Phi$ and $\mathbf{\Sigma}_j^\Phi$ are not only the same but are in the same order, $\mathbf{v}_{i_k}^\phi = \mathbf{v}_{j_k}^\phi$. Using these two results, we can write

$$Q_1 \leq \frac{\sum_{m=1}^{p}\lambda_{i_m}^\phi\lambda_{j_m}^\phi}{\sum_{m=1}^{p}\lambda_{i_m}^{\phi^2}+\sum_{m=1}^{p}\lambda_{j_m}^{\phi^2}}.$$



(a) (b) (c) (d)

Fig. 2. Three examples of the use of the homoscedastic criterion, Q_1. The examples are for two Normal distributions with equal covariance matrix up to scale and rotation. (a) The value of Q_1 decreases as the angle θ increases. The 2D rotation between the two distributions θ is in the x axis. The value of Q_1 is in the y axis. (b) When $\theta = 0^o$, the two distributions are homoscedastic, and Q_1 takes its maximum value of .5. Note how for distributions that are close to homoscedastic (i.e., $\theta \approx 0^o$), the value of the criterion remains high. (c) When $\theta = 45^o$, the value has decreased about .4. (d) By $\theta = 90^o$, $Q_1 \approx .3$.

Now, let us define every eigenvalue of $\mathbf{\Sigma}_i^\Phi$ as a multiple of those of $\mathbf{\Sigma}_j^\Phi$, i.e., $\lambda_{i_m}^\phi = k_m \lambda_{j_m}^\phi$, $k_m \geq 0$, $\forall m = 1, \ldots, p$. This allows us to rewrite our criterion as

$$Q_1 \leq \frac{\sum_{m=1}^p k_m \lambda_{j_m}^{\phi^2}}{\sum_{m=1}^p \lambda_{j_m}^{\phi^2}(k_m^2 + 1)}.$$

From the above equation, we see that $Q_1 \geq 0$, since all its variables are positive. The maximum value of Q_1 will be attained when all $k_m = 1$, which yields $Q_1 = .5$. We now note that having all $k_m = 1$ implies that the eigenvalues of the two covariance matrices are the same. We also know that the maximum of Q_1 can only be reached when the eigenvectors are the same and in the same order, as stated above. This means that the two Normal distributions are homoscedastic in the kernel space defined by $\phi(.)$ when $Q_1 = .5$. ■

From the above result, we see that we can already detect when two distributions are homoscedastic in a kernel space. This means that for a given kernel function, we can find those kernel parameters which give us $Q_1 = .5$. Note that the closer we get to this maximum value, the more similar the two distributions ought to be, since their eigenvalues will become closer to each other. To show this, we would now like to prove that when the value of Q_1 increases, then the divergence between the two distributions decreases.

Divergence is a classical mechanism used to measure the similarity between two distributions. A general type of divergence employed to calculate the similarity between samples from convex sets is the Bregman divergence [3]. Formally, for a given continuously-differentiable strictly convex function $G : \mathbb{R}^{p\times p} \to \mathbb{R}$, the Bregman divergence over real symmetric matrices is defined as

$$\mathbb{B}_G(\mathbf{X}, \mathbf{Y}) = G(\mathbf{X}) - G(\mathbf{Y}) - tr(\nabla G(\mathbf{Y})^T(\mathbf{X} - \mathbf{Y})), \quad (2)$$

where $\mathbf{X}, \mathbf{Y} \in \{\mathbf{Z} \,|\, \mathbf{Z} \in \mathbb{R}^{p\times p}$, and $\mathbf{Z} = \mathbf{Z}^T\}$, and ∇ is the gradient.

Note that the definition given above for the Bregman divergence is very general. In fact, many other divergence measures (such as the Kullback-Leibler) as well as several commonly employed distances (e.g. Mahalanobis and Frobenius) are a particular case of Bregman's. Consider the case where $G(\mathbf{X}) = tr(\mathbf{X}^T\mathbf{X})$, which computes the trace of the covariance matrix, i.e., the Frobenius norm. In this case, the Bregman divergence is $\mathbb{B}_G(\mathbf{\Sigma}_1, \mathbf{\Sigma}_2) = tr(\mathbf{\Sigma}_1^2) + tr(\mathbf{\Sigma}_2^2) - 2tr(\mathbf{\Sigma}_1\mathbf{\Sigma}_2)$, where, as above, $\mathbf{\Sigma}_i$ are the covariance matrices of the two distributions that we wish to compare. We can also rewrite this result using the covariances in the kernel space as,

$$\mathbb{B}_G(\mathbf{\Sigma}_1^\Phi, \mathbf{\Sigma}_2^\Phi) = tr(\mathbf{\Sigma}_1^{\Phi^2}) + tr(\mathbf{\Sigma}_2^{\Phi^2}) - 2tr(\mathbf{\Sigma}_1^\Phi\mathbf{\Sigma}_2^\Phi),$$

where now $G(\mathbf{X}) = tr(\Phi(\mathbf{X})^T\Phi(\mathbf{X}))$.

Note that to decrease the divergence (i.e., the value of $\mathbb{B}_G$), we need to minimize $tr(\mathbf{\Sigma}_1^{\Phi^2}) + tr(\mathbf{\Sigma}_2^{\Phi^2})$ and/or maximize $tr(\mathbf{\Sigma}_1^\Phi\mathbf{\Sigma}_2^\Phi)$. The more we lower the former and increase the latter, the smaller the Bregman divergence will be. Similarly, when we decrease the value of $tr(\mathbf{\Sigma}_1^{\Phi^2}) + tr(\mathbf{\Sigma}_2^{\Phi^2})$ and/or increase that of $tr(\mathbf{\Sigma}_1^\Phi\mathbf{\Sigma}_2^\Phi)$, we make the value of Q_1 larger. Hence, as the value of our criterion Q_1 increases, the Bregman divergence between the two distributions decreases, i.e., the two distributions become more alike. This result is illustrated in Fig. 2. We can formally summarize this result as follows.

Theorem 2: Maximizing Q_1 is equivalent to minimizing the Bregman divergence $\mathbb{B}_G(\mathbf{\Sigma}_1^\Phi, \mathbf{\Sigma}_2^\Phi)$ between the two kernel covariance matrices $\mathbf{\Sigma}_1^\Phi$ and $\mathbf{\Sigma}_2^\Phi$, where $G(\mathbf{X}) = tr(\Phi(\mathbf{X})^T\Phi(\mathbf{X}))$.

We have now shown that the criterion Q_1 increases as any two distributions become more similar to one another. We can readily extend this result to the multiple distribution case,

$$Q_1(\phi) = \frac{2}{C(C-1)} \sum_{i=1}^{C-1} \sum_{k=i+1}^{C} \frac{tr(\mathbf{\Sigma}_i^\Phi\mathbf{\Sigma}_k^\Phi)}{tr(\mathbf{\Sigma}_i^{\Phi^2}) + tr(\mathbf{\Sigma}_k^{\Phi^2})}, \quad (3)$$

where $\mathbf{\Sigma}_i^\Phi$ is the sample covariance matrix of the i^{th} class. This criterion measures the average homoscedasticity of all pairwise class distributions.

This criterion can be directly used in KDA, KNDA and others. Moreover, the same criterion can be readily extended to work in KSDA,

$$Q_1(\phi, H_1, \ldots, H_C) = \frac{1}{h} \sum_{i=1}^{C-1} \sum_{j=1}^{H_i} \sum_{k=i+1}^{C} \sum_{l=1}^{H_k} \frac{tr(\mathbf{\Sigma}_{ij}^\Phi\mathbf{\Sigma}_{kl}^\Phi)}{tr(\mathbf{\Sigma}_{ij}^{\Phi^2}) + tr(\mathbf{\Sigma}_{kl}^{\Phi^2})},$$

where $\mathbf{\Sigma}_{ij}^\Phi$ is the sample covariance matrix of the j^{th} subclass of class i, and h is the number of summing terms.

The reason we needed to derive the above criterion is because, in the multi-class case, the addition of the Bregman divergences would cancel each other out. Moreover, the derived criterion is scale invariant, while Bregman is not.

It may now seem that the criterion Q_1 is ideal for all kernel versions of DA. To study this further, let us define a particular kernel function. An appropriate kernel is the RBF (Radial

Basis Function), because it is specifically tailored for Normal distributions. We will now show that, although homoscedasticity guarantees that the Bayes classifier is linear in this RBF kernel space, it does not guarantee that the class distributions will be separable. In fact, it can be shown that Q_1 may favor a kernel map where all (sub)class distributions become the same, i.e., identical covariance matrix and mean. Indeed a particular but useless case of homoscedasticity in classification problems.

Theorem 3: The RBF kernel is $k(\mathbf{x}_i,\mathbf{x}_j)=\exp\left(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{\sigma}\right)$, with scale parameter σ. In the two class problem, $C=2$, let the pairwise between class distances be $\{D_{11},D_{12},\ldots,D_{n_1n_2}\}$, where $D_{ij}=\|\mathbf{x}_i-\mathbf{x}_j\|_2^2$ is the (squared) Euclidean distance calculated between two sample vectors, $\mathbf{x}_i$ and $\mathbf{x}_j$, of different classes, and n_1 and n_2 are the number of elements in each class. Similarly, let the pairwise within class distances be $\{d_{11}^1,d_{12}^1,\ldots,d_{n_1n_1}^1,d_{11}^2,d_{12}^2,\ldots,d_{n_2n_2}^2\}$, where $d_{kl}^c=\|\mathbf{x}_{ck}-\mathbf{x}_{cl}\|_2^2$ is the Euclidean distances between sample vectors of the same class c. And, use $\mathcal{S}_W$ with the normalized weights

$$\gamma_{ckl}=\frac{\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)}{\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)}$$

and $\mathcal{S}_B$ with the normalized weights

$$\rho_{1i2j}=\frac{\exp\left(-\frac{2D_{ij}}{\sigma}\right)}{\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\exp\left(-\frac{2D_{ij}}{\sigma}\right)}.$$

Then, if $tr(\mathcal{S}_B)>tr(\mathcal{S}_W)$, $Q_1(.)$ monotonically increases with σ, i.e., $\frac{\partial Q_1}{\partial\sigma}\geq 0$.

Proof: Note that both of the numerator and denominator of Q_1 can be written in the form of $\sum_i\sum_j\exp\left(-2\|\mathbf{x}_i-\mathbf{x}_j\|_2^2/\sigma\right)$. Its partial derivative with respect to σ is, $\sum_i\sum_j\frac{2\|\mathbf{x}_i-\mathbf{x}_j\|_2^2}{\sigma^2}\exp\left(-2\|\mathbf{x}_i-\mathbf{x}_j\|_2^2/\sigma\right)$. Substituting for D_{ij} and d_{kl}, we have $\frac{\partial Q_1}{\partial\sigma}$ equal to

$$\frac{\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\exp\left(-\frac{2D_{ij}}{\sigma}\right)\frac{2D_{ij}}{\sigma^2}\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)}{\left[\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)\right]^2}$$
$$-\frac{\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)\frac{2d_{kl}^c}{\sigma^2}\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\exp\left(-\frac{2D_{ij}}{\sigma}\right)}{\left[\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)\right]^2}$$

We want to know when $\partial Q_1/\partial\sigma\geq 0$, which is the same as

$$\frac{\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\exp\left(-\frac{2D_{ij}}{\sigma}\right)D_{ij}}{\sum_{i=1}^{n_1}\sum_{j=1}^{n_2}\exp\left(-\frac{2D_{ij}}{\sigma}\right)}$$
$$>\frac{\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)d_{kl}^c}{\sum_{c=1}^{2}\sum_{k=1}^{n_c}\sum_{l=1}^{n_c}\exp\left(-\frac{2d_{kl}^c}{\sigma}\right)}$$

The left hand side of this inequality is the estimate of the between class variance, while the right hand side is the within class variance estimate, since D_{ij} and d_{ij}^c can be rewritten as the trace of the outer product $tr((\mathbf{x}_i-\mathbf{x}_j)(\mathbf{x}_i-\mathbf{x}_j)^T)$. Substituting for the above defined γ_{ckl} and ρ_{1i2j}, we have $\partial Q_1/\partial\sigma\geq 0$ when $tr(\mathcal{S}_B)>tr(\mathcal{S}_W)$. ∎

This latest theorem shows that when σ approaches infinity, $\frac{\partial Q_1}{\partial\sigma}$ approaches zero and, hence, Q_1 tends to its maximum value of .5. Increasing σ to infinity in the RBF kernel will result in a space where the two class distributions become identical. This will happen whenever $tr(\mathcal{S}_B)>tr(\mathcal{S}_W)$. This is a fundamental theorem of DA because it shows the relation between KDA, the weighted LDA version of [20] and the NDA method of [11]. Theorem 3 shows that these variants of DA are related to the idea of maximizing homoscedasticity as defined in this paper. It also demonstrates the importance of the metrics in weighed LDA and NDA. In particular, the above result proves that if, after proper normalization, the between class differences are larger than the within class differences, then classification in the kernel space optimized with Q_1 will be as bad as random selection. One indeed wants the class distributions to become homoscedastic in the kernel space, but not at the cost of classification accuracy, which is the underlying goal.

To address the problem outlined in Theorem 3, we need to consider a second criterion which is directly related to class separability. Such a criterion is simply given by the trace of the between-class (or -subclass) scatter matrix, since this is proportional to class separability, $Q_2(\phi)=$

$$\begin{aligned}tr\left(\mathbf{S}_B^{\Phi}\right) &= tr\left(\sum_{i=1}^{C-1}\sum_{k=i+1}^{C}p_ip_k(\mu_i^{\phi}-\mu_k^{\phi})(\mu_i^{\phi}-\mu_k^{\phi})^T\right)\\ &= \sum_{i=1}^{C-1}\sum_{k=i+1}^{C}p_ip_k\|\mu_i^{\phi}-\mu_k^{\phi}\|^2. \qquad (4)\end{aligned}$$

Again, we can readily extend this result to work with subclasses,

$$\begin{aligned}Q_2(\phi,H_1,\ldots,H_C) &= tr\left(\mathbf{\Sigma}_B^{\Phi}\right)\\ &= \sum_{i=1}^{C-1}\sum_{j=1}^{H_i}\sum_{k=i+1}^{C}\sum_{l=1}^{H_k}p_{ij}p_{kl}\|\mu_{ij}^{\phi}-\mu_{kl}^{\phi}\|^2.\end{aligned}$$

Since we want to maximize homoscedasticity and class separability, we need to combine the two criteria of (3) and (4),

$$Q(.)=Q_1(.)\,Q_2(.). \qquad (5)$$

The product given above is an appropriate way to combine independent measures of different magnitude as is the case with Q_1 and Q_2.

Using the criterion given in (5), the optimal kernel function, ϕ^*, is

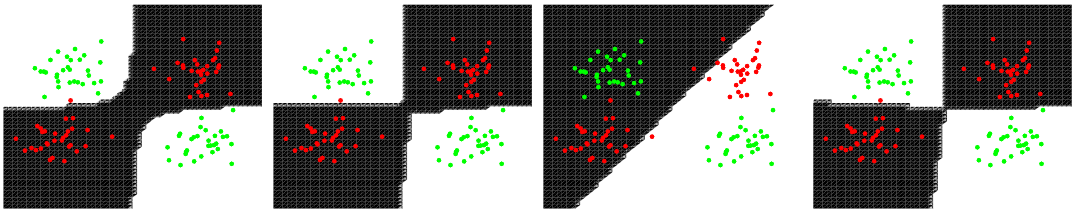
$$\phi^*=\arg\max_{\phi}Q(\phi).$$

In KSDA, we optimize the number of subclasses and the kernel as

$$\phi^*,H_1^*,\ldots,H_C^*=\arg\max_{\phi,H_1,\ldots,H_C}Q(\phi,H_1,\ldots,H_C).$$

Also, recall that in KSDA (as in SDA), we need to divide the data into subclasses. As stated above we assume that the underlying class distribution can be approximated by a mixture of Gaussians. This assumption, suggests the following ordering of the samples: $\widehat{\mathbf{X}}=\{\hat{\mathbf{x}}_1,\ldots,\hat{\mathbf{x}}_n\}$, where $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_n$ are the two most dissimilar feature vectors and $\hat{\mathbf{x}}_k$ is the $k-1^{th}$ feature vector closest to $\hat{\mathbf{x}}_1$. This ordering allows us to divide the set of samples into H subgroups, by simply dividing $\widehat{\mathbf{X}}$ into H parts. This approach has been shown to be appropriate for finding subclass divisions [35].

As a final note, it is worth emphasizing that, as opposed to CV, the derived criterion will use the whole data in the training set for estimating the data distributions because there is no need for a verification set. With a limited number of



(a) (b) (c) (d)

Fig. 3. Here we show a two class classification problem with multi-modal class distributions. When $\sigma = 1$ both KDA (a) and KSDA (b) generate solutions that have small training error. (c) However, when the model complexity is small, $\sigma = 3$, KDA fails. (d) KSDA's solution resolves this problem with piecewise smooth, nonlinear classifiers.

training samples, this will generally yield better estimates of the unknown underlying distribution. The other advantage of the derived approach is that it can be optimized using gradient descent, by taking $\partial Q(k(\mathbf{x}_i, \mathbf{x}_j))/\partial\sigma$. In particular, we employ a quasi-Newton approach with a Broyden-Fletcher-Goldfarb-Shanno Hessian update [6], yielding a fast convergence. The derivations are in the Supplementary File. The initial value for the kernel parameter is set to be the mean of the distances between all pairwise training samples.

IV. Generalization

A major goal in pattern recognition is to find classification criteria that have small generalization error, i.e., small expected error on the unobserved data. This mainly depends on the number of samples in our training set, training error and the model (criterion) complexity [14]. Since the training set is usually fixed, we are left to select a proper model. Smooth (close to linear) classifiers have a small model complexity but large training error. On the other hand wiggly classifiers may have a small training error but large model complexity. To have a small generalization error, we need to select a model that has moderate training error and model complexity. Thus, in general, the simpler your classifier is, the smaller the generalization error. However, if the classifier is too simple, the training error may be very large.

KDA is limited in terms of model complexity. This is mainly because KDA assumes each class is represented with unimodal distributions. If there is a multimodal structure in each class, KDA would select wiggly functions in order to minimize the classification error. To avoid this, the model complexity may be limited to smooth solutions, which would generally result in large training errors and, hence, large generalization errors.

This problem can be solved by using an algorithm that considers multimodal class representations, e.g., KSDA. While KDA can find wiggly functions to separate multimodal data, KSDA can find several functions which are smoother and carry smaller training errors. We can illustrate this theoretical advantage of KSDA with a simple 2-class classification example, Fig. 3. In this figure, each class consists of 2 nonlinearly separable subclasses. Fig. 3(a) shows the solution of KDA obtained with the RBF kernel with $\sigma = 1$. Fig. 3(b) shows the KSDA solution. KSDA can obtain a classification function that has the same training error with smaller model complexity, i.e., smoother classification boundaries. When we reduce the model complexity by increasing σ to 3, KDA leads to a large training error, Fig. 3(c). This does not occur in KSDA, Fig. 3(d). A similar argument can be used to explain the problems faced with Maximum Likelihood (ML) classification when modeling the original data as a Mixture of Gaussians (MoG) in the original space. Unless one has access to a sufficiently large set (i.e., proportional to the number of dimensions of this *original* feature space), the results will not generalize well. In the section to follow, we will show experimental results with real data that verify the observation discussed in this section.

V. Experimental Results

In this section, we will use our criterion to optimize the kernel parameter of KDA, KNDA and KSDA. We will give comparative results with CV, the Fisher criterion of [31] the use of the Bregman divergence, and to other nonlinear methods – Kernel PCA (KPCA), HLDA and LPP – and related linear approaches – LDA, NDA, RDA, SDA, and aPAC. The dimensionality of the reduced space is taken to be the rank of the matrices used by the DA approach and to keep 90% of the variance in PCA and KPCA. We also provide comparisons with Kernel Support Vector Machines (KSVM) [29] and the use of ML in MoG [24], two classical alternatives for nonlinear classification.

A. Databases and notation

The first five data-sets are from the UCI repository [2]. The Monk problem is given by a 6-dimensional feature space defining six joints of a robot and two classes. Three different case scenarios are considered, denoted Monk 1, 2 and 3. The Ionosphere set corresponds to satellite imaging for the detection of two classes (structure or not) in the ground. And, in the NIH Pima set, the goal is to detect diabetes from eight measurements.

We also use the ETH-80 [18] database. It includes a total of $3,280$ images of the following 8 categories: apples, pears, cars, cows, horses, dogs, tomatoes and cups. Each category includes 10 objects (e.g., ten apples). Each of the (80) objects has been photographed from 41 orientations. We resized all the images to 25×30 pixels. The pixel values in their vector form ($\mathbf{x} \in \mathbb{R}^{750}$) are used in the appearance-based recognition approach. As it is typical in this database, we will use the leave-one-object-out test. That is, the images of 79 objects are used for training, those of the remaining object for testing. We test all options and calculate the average recognition rate.

We also use 100 randomly selected subjects from the AR face database [21]. All images are first aligned with respect to their eyes, mouth and jaw line before cropping and resizing them to a standard size of 29×21 pixels. This database contains images of two different sessions, each taken two weeks apart. The images in the first and second session contain the same facial expressions and occlusions and were taken under the same illumination conditions. We use the images in the first session for training and those in the second session for testing.

We also use the Sitting Posture Distribution Maps data-set (SPDM) of [36]. Here, samples were collected using a chair equipped with a pressure sensor sheet located on the sit-pan and back-rest. The pressure maps provide a total of $1,280$ pressure values. The database includes samples of 50 individuals. Each participant provided five samples of each of the ten different postures. Our goal is to classify each of the samples into one of the ten siting postures. This task is made difficult by the nonparametric nature of the samples in each class [36]. We randomly selected 3 samples from each individual and posture for training, and used the rest for testing.

The Modified National Institute of Standards and Technology (MNIST) database of [17] is a large collection of various sets of handwritten digit (0-9). The training set consists of 60,000 samples. The test set has 10,000 samples. All the digits have been size-normalized to 28×28. We randomly select 30,000 samples for training, with 3,000 samples in each class. This is done to reduce the size of the Gram matrix, allowing us to run the algorithm on a desktop.

As defined above, we employe the RBF kernel. The kernel parameter σ in KPCA is optimized with CV. CV is also used in KDA, KNDA and KSDA, denoted: KDA$_{CV}$, KNDA$_{CV}$ and KSDA$_{CV}$. The kernel parameter is searched in the range $[m-2st, m+2st]$, where m and st are the mean and standard deviation of the distances between all pairwise training samples. We use 10-fold cross validation in the UCI data-sets and 5-fold cross validation in the others. In KNDA and KSDA, the number of nearest neighbors and subclasses are also optimized. In KSDA, we test partitions from 1 to 10 subclasses. We also provide comparative results when optimizing σ with the approach of [31], denoted: KDA$_F$, KNDA$_F$ and KSDA$_F$. The two parameters of LPP (i.e., the number of nearest neighbors, and the heat kernel) are optimized with CV. The DA algorithms with our Homoscedastic-based optimization will be denoted: KDA$_H$, KNDA$_H$ and KSDA$_H$. The same algorithms optimized using Bregman are denoted: KDA$_B$, KNDA$_B$ and KSDA$_B$.

B. Results

The algorithms summarized above are first employed to find the subspace where the feature vectors of different classes are most separated according to the algorithm's criterion. In the reduced space we employ a variety of classification methods.

In our first experiment, we use the nearest mean (NM) classifier. The NM is an ideal classifier because it provides the *Bayes optimal* solution whenever the class distributions are homoscedastic Gaussians [10]. Thus, the results obtained with the NM will illustrate whether the derived criterion has achieved the desirable goal. The results are shown in Table I. We see that the kernel algorithms optimized with the proposed Homoscedastic-based criterion generally obtain higher classification rates. To further illustrate this point, the table includes a rank of the algorithms following the approach of [5]. As predicted by our theory, the additional flexibility of KSDA allows it to achieve the best results.

Our second choice of classifier is the classical nearest neighbor (NN) algorithm. Its classification error is known to be less than twice the Bayes error. This makes it appropriate for the cases where the class distributions are not homoscedastic. These results are in Table II. A recently proposed classification algorithm [26] emphasizes smoother classification boundaries in the NN framework. This algorithm is based on the approximation of the nonlinear decision boundary using the sample points closest to the classification boundary. The classification boundary is smoothed using Tikhonov regularization. Since our criterion is used to make the classifier in the kernel space as linear as possible, smooth (close to linear) classifiers are consistent with this goal and should generally lead to better results. We present the results obtained with this alternative approach in Table III.

Finally, recall that the goal of the Homoscedastic criterion is to make the Bayes classifier in the kernel space linear. If this goal were achieved, one would expect a linear classifier such as linear Support Vector Machines (SVM) to yield good classification results in the corresponding subspace. We verified this hypothesis in our final experiment, Table IV.

Note that regardless of the classifier used in the reduced space, KSDA$_H$ consistently yields the best results. It is followed by KDA$_H$ and KSVM. See the supplementary file for details.

As mentioned earlier, the advantage of the proposed criterion is not only that it achieves higher classification rates, but that it does so at a lower computational cost, Table V. Note that the proposed approach generally reduces the running time by one order of magnitude.

TABLE I

RECOGNITION RATES (IN PERCENTAGES) WITH NEAREST MEAN

DATA SET	KSDA$_H$	KSDA$_F$	KSDA$_B$	KSDA$_{CV}$	KDA$_H$	KDA$_F$	KDA$_B$	KDA$_{CV}$	KNDA$_H$
ETH-80	**82.6***	73.5	61.7	77.4	**82.6***	81.6	61.7	71.6	**76.2**
AR DATABASE	**88.1***	78.2	65.5	84.2	**87.5***	86.7	69.5	84.2	71.3
SPDM	**84.6***	80.1	67.9	**83.9***	**84.6***	83.2	67.9	83.3	**82.4**
MONK1	**88.2***	85.0	71.1	**88.0***	84.0	**89.6***	65.3	83.1	**70.1**
MONK2	76.6	**82.2***	56.7	74.5	**80.1**	75.2	55.6	70.1	**73.5**
MONK3	**96.3***	88.7	85.4	94.0	**93.1**	89.7	85.7	82.4	67.6
IONOSPHERE	93.4	84.8	88.1	**96.0***	**93.4**	86.1	67.6	80.8	74.8
PIMA	**80.4***	77.4	70.2	**80.4***	**78.6**	75.0	75.0	72.6	65.5
MNIST	**98.0***	96.9	92.0	97.4	**98.1***	96.6	92.0	97.2	94.6
RANK	1.9*	7.0	13.3	3.6	2.8	5.4	14.2	9.2	12.2

DATA SET	KNDA$_F$	KNDA$_B$	KNDA$_{CV}$	MOG	KSVM	KPCA	PCA	LDA	HLDA
ETH-80	74.6	65.6	73.6	69.2	81.8	56.9	56.5	63.3	58.2
AR DATABASE	61.4	72.5	**74.3**	75.5	86.7	42.2	24.0	79.3	67.4
SPDM	**82.9**	53.4	75.6	73.4	84.7*	62.6	66.4	44.5	68.0
MONK1	65.7	50.0	63.4	80.3	83.6	67.4	66.0	64.6	66.2
MONK2	64.8	61.8	71.8	75.9	82.6*	53.7	53.5	55.1	53.5
MONK3	63.7	**77.8**	66.4	89.4	93.5	78.9	80.6	63.9	81.3
IONOSPHERE	62.3	65.6	**78.2**	82.1	96.0*	89.4	62.3	57.0	90.1
PIMA	67.3	**70.8**	66.7	75.0	79.2	50.0	56.0	61.3	77.4
MNIST	94.3	93.1	**96.4**	88.6	97.6	80.6	82.2	86.7	85.5
RANK	14.7	15.8	13.3	9.8	2.7	18.0	19.1	18.3	14.1

Note that the results obtained with the Homoscedastic criterion are generally better than those given by the Fisher, Bregman and CV criteria. The best of the three results in each of the discriminant methods is bolded. The symbol * is used to indicate the top result among all algorithms. Rank goes from smallest (best) to largest.

TABLE II

RECOGNITION RATES (%) WITH NEAREST NEIGHBOR

DATA SET	KSDA$_H$	KSDA$_F$	KSDA$_B$	KSDA$_{CV}$	KDA$_H$	KDA$_F$	KDA$_B$	KDA$_{CV}$	KNDA$_H$
ETH-80	**82.8***	73.6	62.3	76.8	**82.8***	81.0	62.3	71.6	**76.2**
AR DATABASE	**96.7***	78.3	66.9	84.2	**88.3**	87.5	71.3	84.2	**69.2**
SPDM	**84.9***	80.1	68.2	83.7	**84.9***	**84.2**	68.2	83.3	73.9
MONK1	**89.1***	84.5	78.2	87.5	84.3	**89.6***	72.5	83.1	**78.2**
MONK2	77.8	83.1	**86.1***	75.7	**80.1**	75.2	77.6	70.1	**85.0***
MONK3	**94.4***	87.7	81.5	89.8	**93.5**	88.0	89.4	82.4	**82.1**
IONOSPHERE	**94.4**	84.8	91.4	**94.0**	**94.4**	86.5	70.9	80.8	87.4
PIMA	75.0	73.8	66.7	**76.8**	70.2	69.8	64.9	**72.6**	67.3
MNIST	**97.8***	96.9	91.8	97.2	**97.2**	**97.1**	**91.8**	96.7	**95.6**
RANK	2.9*	8.0	13.6	5.3	3.7	7.7	15.4	10.8	11.3

DATA SET	KNDA$_F$	KNDA$_B$	KNDA$_{CV}$	MOG	KSVM	KPCA	PCA	LDA	HLDA
ETH-80	74.6	68.0	70.6	69.2	81.8	62.2	64.3	64.3	56.5
AR DATABASE	64.2	**70.6**	**70.2**	75.5	86.7	42.5	58.6	77.7	67.5
SPDM	**75.6**	33.5	70.3	73.4	84.7	75.0	81.5	66.5	65.3
MONK1	77.1	74.5	72.2	80.3	83.6	90.3*	81.3	69.0	84.2
MONK2	81.0	79.9	78.5	75.9	82.6	68.3	66.7	67.4	83.6
MONK3	81.3	77.6	80.3	89.4	93.5	87.8	87.3	70.6	84.5
IONOSPHERE	86.1	**90.1**	86.1	82.1	96.0*	89.4	92.1	74.8	88.7
PIMA	67.3	66.1	**69.1**	75.0	79.2*	56.0	64.3	57.7	68.5
MNIST	**95.4**	92.1	**95.5**	88.6	97.6	94.1	90.2	89.8	80.6
RANK	12.7	15.7	14.1	12.6	3.2	14.2	14.3	18.4	14.3

VI. CONCLUSIONS

This paper has presented the derivations of a first approach to optimize the parameters of a kernel whose function is to map the

TABLE III

RECOGNITION RATES (%) WITH THE CLASSIFICATION METHOD OF [26]

DATA SET	KSDA$_H$	KSDA$_F$	KSDA$_B$	KSDA$_{CV}$	KDA$_H$	KDA$_F$	KDA$_B$	KDA$_{CV}$	KNDA$_H$
ETH-80	**83.5***	73.9	62.3	76.4	**83.5***	82.8	62.3	72.9	**76.2**
AR DATABASE	**96.6***	78.5	66.9	85.1	**90.6**	86.7	71.3	85.1	70.9
SPDM	**84.3***	75.3	68.2	**83.9***	**84.3***	83.4	68.2	82.6	75.6
MONK1	**90.2***	76.6	71.5	82.9	**89.6**	87.7	72.2	88.7	**65.2**
MONK2	**83.3***	77.5	60.6	75.7	80.6	**82.9**	73.8	78.5	**74.1**
MONK3	**94.6***	83.3	86.1	86.3	**93.5**	92.4	89.4	91.2	68.5
IONOSPHERE	**94.3**	84.8	84.8	86.1	**94.3**	86.8	80.1	86.8	80.8
PIMA	**80.4***	76.8	79.2	76.2	**78.6**	73.0	64.9	69.0	**72.0**
MNIST	**97.8***	96.9	91.8	97.3	**97.2**	**97.2**	91.8	96.7	**95.6**
RANK	1.2*	9.4	14.4	6.7	2.7	4.6	15	6.9	14.2

DATA SET	KNDA$_F$	KNDA$_B$	KNDA$_{CV}$	MOG	KSVM	KPCA	PCA	LDA	HLDA
ETH-80	74.2	68.2	71.2	69.2	81.8	60.3	67.1	64.3	59.1
AR DATABASE	63.2	70.6	**72.6**	75.5	86.7	49.5	44.5	70.9	67.5
SPDM	**77.9**	35.6	71.5	73.4	84.7*	75.1	77.0	56.2	53.4
MONK1	62.0	61.4	62.3	80.3	83.6	77.3	78.2	67.4	71.5
MONK2	64.8	62.3	56.9	75.9	82.6	58.6	56.7	70.6	58.3
MONK3	64.8	**85.4**	66.2	89.4	93.5	91.2	89.7	70.8	93.8
IONOSPHERE	**82.8**	77.5	78.1	82.1	96.0*	82.1	82.1	74.8	94.0
PIMA	67.9	69.0	67.9	75.0	79.2	60.7	70.2	57.7	72.6
MNIST	95.4	92.1	**95.6**	88.6	97.6	94.1	90.2	89.8	82.7
RANK	14.7	17.6	16.1	11.6	2.7	15.6	14.6	17.8	14.9

TABLE IV

RECOGNITION RATES (%) WITH LINEAR SVM

DATA SET	KSDA$_H$	KSDA$_F$	KSDA$_B$	KSDA$_{CV}$	KDA$_H$	KDA$_F$	KDA$_B$	KDA$_{CV}$	KNDA$_H$
ETH-80	**83.0***	73.6	61.9	77.4	**83.0***	82.2	61.9	71.3	**75.6**
AR DATABASE	**88.1***	79.6	65.5	83.1	**87.5***	86.7	69.5	83.1	**79.4**
SPDM	82.1	**84.6***	67.5	82.3	82.1	**83.6**	67.5	82.6	82.2
MONK1	**89.1***	88.2	50.0	86.1	84.7	**89.7***	52.1	86.1	**69.9**
MONK2	77.1	**81.5**	67.1	73.8	**80.1**	75.2	67.1	75.1	67.1
MONK3	**95.6***	91.9	47.2	94.4	**92.8**	89.1	47.2	81.5	**81.7**
IONOSPHERE	93.4	86.1	82.1	**96.7***	**93.4**	86.1	82.1	82.1	**82.1**
PIMA	**79.8***	78.6	64.9	**79.8***	**78.0***	75.0	64.3	**72.8**	64.3
MNIST	**97.9**	96.9	92.0	97.3	**98.1***	96.7	92.0	97.2	94.7
RANK	2.8*	5.6	17.8	4.3	4.1	5.8	17.7	9.5	11.9

DATA SET	KNDA$_F$	KNDA$_B$	KNDA$_{CV}$	MOG	KSVM	KPCA	PCA	LDA	HLDA
ETH-80	**75.2**	65.6	74.6	69.2	81.8	65.3	60.1	65.3	68.4
AR DATABASE	75.7	72.5	78.6	75.5	86.7	42.1	66.7	79.3	70.1
SPDM	82.9	52.7	**84.0**	73.4	84.7*	66.7	76.5	50.3	69.3
MONK1	62.5	50.0	63.4	80.3	83.6	88.4*	67.8	65.6	68.5
MONK2	**83.1***	67.1	67.1	75.9	82.6	50.0	67.1	67.1	67.1
MONK3	**81.7**	47.2	81.0	89.4	93.5	94.4	81.3	63.9	81.9
IONOSPHERE	**82.1**	**82.1**	**82.1**	82.1	96.0	82.1	84.8	84.8	93.4
PIMA	64.3	64.3	64.3	75.0	79.2*	64.3	68.6	64.9	76.2
MNIST	94.3	93.3	**96.2**	88.6	97.6	81.0	82.2	87.0	85.5
RANK	11.6	17.3	13.0	11.5	3.3	16.1	16.1	15.8	12.5

original class distributions to a space where these are optimally (wrt Bayes) separated with a hyperplane. We have achieved this by selecting the kernel parameters that make the class Normal distributions most homoscedastic while maximizing class separability. Experimental results in a large variety of datasets has demonstrated that this approach achieves higher recognition rates than most other methods defined to date. We have also shown that adding the subclass divisions to the optimization process (KSDA) allows the DA algorithm to achieve better generalizations. And, we have formally defined the relationship between KDA and other variants of DA, such as weighted DA, NDA and SDA. Extensions to work with very large datasets will be considered in future work.

VII. ACKNOWLEDGMENTS

We thank the reviewers for their constructive comments. This research was partially supported by the National Institutes of Health under grant R01 DC 005241 and by the National Science Foundation under grant IIS 0713055.

TABLE V

TRAINING TIME (IN SECONDS)

DATA SET	KSDA$_H$	KSDA$_{CV}$	KDA$_H$	KDA$_{CV}$	KNDA$_H$	KNDA$_{CV}$	KSVM
ETH-80	7.3×10^4	3.6×10^5	1.8×10^3	9.0×10^4	7.9×10^4	8.5×10^5	1.8×10^4
AR DATABASE	4.2×10^4	3.5×10^5	3.1×10^3	9.0×10^4	1.5×10^4	1.7×10^5	1.2×10^4
SPDM	1.8×10^4	6.5×10^4	1.8×10^2	4.6×10^4	2.1×10^4	1.6×10^5	9.6×10^3
MONK1	4.4	51.3	0.7	6.8	26.4	504.8	3.7
MONK2	4.6	88.1	1.2	11.5	41.3	978.1	17.8
MONK3	3.2	50.7	0.7	6.4	23.1	516.0	2.2
IONOSPHERE	6.6	134.8	1.3	15.7	76.6	1479.5	10.1
PIMA	80.2	2521.7	12.1	380.1	374.4	10889.7	150.6
MNIST	3.6×10^5	2.0×10^6	1.9×10^5	1.1×10^6	3.2×10^5	4.6×10^6	4.5×10^5

REFERENCES

[1] G. Baudat and F. Anouar. Generalized discriminant analysis using a kernel approach. *Neural Computation*, 12(10):2835–2404, 2000.

[2] C. L. Blake and C. J. Merz. UCI repository of machine learning databases. University of California, Irvine, http ://www.ics.uci.edu/mlearn/MLRepository.html, 1998.

[3] L. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *USSR Comp. Mathematics and Mathematical Physics*, 7:200217, 1967.

[4] B. Chen, L. Yuan, H. Liu, and Z. Bao. Kernel subclass discriminant analysis. *Neurocomputing*, 2007.

[5] J. Demsar. Statistical comparisons of classifiers over multiple data sets. *J. Machine Learning Research*, 7:1–30, 2006.

[6] J. Dennis and R. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Englewood Cliffs, NJ: Prentice-Hall, 1983.

[7] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7, 1936.

[8] R. A. Fisher. The statistical utilization of multiple measurements. *Annals of Eugenics*, 8:376–386, 1938.

[9] J. H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84:165–175, 1989.

[10] K. Fukunaga. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press, San Diego, CA, 1990.

[11] K. Fukunaga and J. Mantock. Nonparametric discriminant analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 5:671–678, 1983.

[12] O. C. Hamsici and A. M. Martinez. Bayes optimality in linear discriminant analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:647–657, 2008.

[13] O. C. Hamsici and A. M. Martinez. Rotation invariant kernels and their application to shape analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[14] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag (2^{nd} Edition), New York, NY, 2009.

[15] X. He and P. Niyogi. Locality preserving projections. In *Proc. Advances in Neural Information Processing Systems 16*, 2004.

[16] S. Kim, A. Magnani, and S. Boyd. Optimal kernel selection in kernel fisher discriminant analysis. In *Int. Conf. Machine Learning*, pages 465–472, 2006.

[17] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of IEEE*, 92(11):2278–2324, 1998.

[18] B. Leibe and B. Schiele. Analyzing appearance and contour based methods for object categorization. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2003.

[19] M. Loog and R. P. W. Duin. Linear dimensionality reduction via a heteroscedastic extension of lda: The chernoff criterion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):732–739, 2007.

[20] M. Loog, R. P. W. Duin, and R. Haeb-Umbach. Multiclass linear dimension reduction by weighted pairwise fisher criteria. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(7):762–766, 2001.

[21] A. M. Martinez and R. Benavente. *The AR Face Database*. CVC Technical Report No. 24, June, 1998.

[22] A. M. Martinez and O. C. Hamsici. Who is LB1? discriminant analysis for the classification of specimens. *Pattern Rec.*, 41:3436–3441, 2008.

[23] A. M. Martinez and M. Zhu. Where are linear feature extraction methods applicable? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(12):1934–1944, 2005.

[24] G. McLachlan and K. Basford. *Mixture Models: Inference and applications to clustering*. Marcel Dekker, 1988.

[25] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. Muller. Fisher discriminant analysis with kernels. In *Proc. IEEE Neural Networks for Signal Processing Workshop*, pages 41–48, 1999.

[26] O. Pujol and D. Masip. Geometry-based ensembles: Towards a structural characterization of the classification boundary. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):1140–1146, 2009.

[27] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, 2001.

[28] C. M. Theobald. An inequality for the trace of the product of two symmetric matrices. *Proceedings of the Cambridge Philosophical Society*, 77:256–267, 1975.

[29] V. Vapnik. *Statistical Learning Theory*. John Wiley and Sons, 1998.

[30] G. Wahba. *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, 1990.

[31] L. Wang, K. Chan, P. Xue, and L. Zhou. A kernel-induced space selection approach to model selection in klda. *IEEE Trans. Neural Networks*, 19:2116–2131, 2008.

[32] J. Yang, A. F. Frangi, J. Yang, D. Zhang, and Z. Jin. KPCA plus LDA: A complete kernel fisher discriminant framework for feature extraction and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(2):230–244, 2005.

[33] M.-H. Yang. Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods. In *Proc. IEEE International Conference on Automatic Face and Gesture Recognition*, 2002.

[34] J. Ye, S. Ji, and J. Chen. Multi-class discriminant kernel learning via convex programming. *J. Machine Lear. Res.*, 9:719–758, 2008.

[35] M. Zhu and A. M. Martinez. Subclass discriminant analysis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 28(8):1274–1286, 2006.

[36] M. Zhu and A. M. Martinez. Pruning noisy bases in discriminant analysis. *IEEE Transactions Neural Networks*, 19(1):148–157, 2008.