Probabilistic Class-Specific Discriminant Analysis

Alexandros Iosifidis
Department of Engineering, ECE, Aarhus University, Denmark
Email: alexandros.iosefidi@eng.au.dk

Abstract—In this paper we formulate a probabilistic model for class-specific discriminant subspace learning. The proposed model can naturally incorporate the multi-modal structure of the negative class, which is neglected by existing methods. Moreover, it can be directly used to define a probabilistic classification rule in the discriminant subspace. We show that existing class-specific discriminant analysis methods are special cases of the proposed probabilistic model and, by casting them as probabilistic models, they can be extended to class-specific classifiers. We illustrate the performance of the proposed model, in comparison with that of related methods, in both verification and classification problems.

Index Terms—Class-Specific Discriminant Analysis, Multi-modal data distributions, Verification, Classification.

I. INTRODUCTION

Class-Specific Discriminant Analysis (CSDA) [1], [2], [3], [4], [5], [6], determines an optimal subspace suitable for verification problems, where the objective is the discrimination of the class of interest from the rest of the world. As an example, let us consider the person identification problem, either through face verification [1], or through exploiting movement information [7]. Different from person recognition, which is a multi-class classification problem assigning a sample (facial image or movement sequence) to a class in a pre-defined set of classes (person IDs in this case), person identification discriminates the person of interest from all rest people.

While multi-class discriminant analysis models, like Linear Discriminant Analysis (LDA) and its variants [8], [9], [10], can be applied in such problems, they are inherently limited by the adopted class discrimination definition. That is, the maximal dimensionality of the resulting subspace is restricted by the number of classes, due to the rank of the between-class scatter matrix. In verification problems involving two classes LDA and its variants lead to one-dimensional subspaces. On the other hand, CSDA by expressing class discrimination using the out-of-class and intra-class scatter matrices is able to define subspaces the dimensionality of which is restricted by the the number of samples forming the smallest class (which is usually the class of interest) or the number of original space dimensions. By defining multiple discriminant dimensions, CSDA has been shown to achieve better class discrimination and better performance compared to LDA [2], [3], [5].

While the definition of class discrimination in CSDA and its variants based on the intra-class and out-of-class scatter matrices overcomes the limitations of LDA related to the dimensionality of the discriminant subspace, it overlooks the structure of the negative class. Since in practice samples forming the negative class belong to many classes, different from the positive one, it is expected that they will form subclasses, as illustrated in the example of Figure 1. Class discrimination as defined by CSDA and its variants disregards this structure.

In this paper, we formulate the class-specific discriminant analysis optimization problem based on a probabilistic model that incorporates the above-described structure of the negative class. We show that the optimization criterion used by standard CSDA and its variants corresponds to a special case of the proposed probabilistic model, while new discriminant subspaces can be obtained by allowing samples of the negative class to form subclasses automatically determined by applying (unsupervised) clustering techniques on the negative class data. Moreover, the use of the proposed probabilistic model for class-specific discriminant learning naturally leads to a classification rule in the discriminant subspace, something that is not possible when the standard CSDA criterion is considered.

II. RELATED WORK

Let us denote by $S_p = \{x_1, \ldots, x_{N_p}\}$ a set of $N_p$ $D$-dimensional vectors representing samples of the positive class and by $S_n = \{x_{N_p+1}, \ldots, x_N\}$, where $N = N_p + N_n$, a set of $N_n$ vectors representing samples of the negative class. In the following we consider the linear class-specific subspace learning case and we will describe how to perform nonlinear (kernel-based) class-specific subspace learning following the same processing steps in Section III-C. We would like determine a linear projection $W \in \mathbb{R}^{D \times d}$, mapping $x_i$ to
Class-specific Discriminant Analysis \[ ^2 \] defines the projection matrix \( \mathbf{W} \) as the one maximizing the following criterion:

\[
\mathcal{J}(\mathbf{W}) = \frac{D_n(\mathbf{W})}{D_p(\mathbf{W})},
\]

where \( D_n(\mathbf{W}) \) and \( D_p(\mathbf{W}) \) are the out-of-class and intra-class distances defined as follows:

\[
D_n(\mathbf{W}) = \sum_{\mathbf{x}_i \in \mathcal{S}_n} \| \mathbf{W}^T \mathbf{x}_i - \mathbf{W}^T \mathbf{m} \|^2 \quad (2)
\]

and

\[
D_p(\mathbf{W}) = \sum_{\mathbf{x}_i \in \mathcal{S}_p} \| \mathbf{W}^T \mathbf{x}_i - \mathbf{W}^T \mathbf{m} \|^2. \quad (3)
\]

\( \mathbf{m} \) is the mean vector of the positive class, i.e. \( \mathbf{m} = \frac{1}{N_p} \sum_{\mathbf{x}_i \in \mathcal{S}_p} \mathbf{x}_i \). That is, it is assumed that the positive class is unimodal and \( \mathbf{W} \) is defined as the matrix mapping the positive class vectors as close as possible to the positive class mean vector, while it maps the negative class vectors as far as possible from it, in the reduced dimensionality space \( \mathbb{R}^d \).

The criterion \( \mathcal{J}(\mathbf{W}) \) in (1) can be expressed as:

\[
\mathcal{J}(\mathbf{W}) = \frac{\text{Tr} (\mathbf{W}^T \mathbf{S}_n \mathbf{W})}{\text{Tr} (\mathbf{W}^T \mathbf{S}_p \mathbf{W})}, \quad (4)
\]

where \( \text{Tr} (\cdot) \) is the trace operator. \( \mathbf{S}_n \in \mathbb{R}^{D \times D} \) and \( \mathbf{S}_p \in \mathbb{R}^{D \times D} \) are the out-of-class and intra-class scatter matrices defined in the space \( \mathbb{R}^D \):

\[
\mathbf{S}_n = \sum_{\mathbf{x}_i \in \mathcal{S}_n} (\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T \quad (5)
\]

and

\[
\mathbf{S}_p = \sum_{\mathbf{x}_i \in \mathcal{S}_p} (\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T. \quad (6)
\]

\( \mathbf{W} \) is obtained by solving the generalized eigen-analysis problem of \( \mathbf{S}_n \mathbf{w} = \lambda \mathbf{S}_p \mathbf{w} \) and keeping the eigenvectors corresponding to the \( d \) largest eigenvalues \([11]\). Here, we should note that, since the rank of \( \mathbf{S}_p \) is at most \( N_p - 1 \) (and considering that usually \( N_n > N_p \)), the dimensionality of the learnt subspace \( d \) is restricted by the number of samples forming the positive class (or by the dimensionality of the original space \( D \)), i.e. \( d \leq \min(N_p - 1, D) \). In the case where \( \mathbf{S}_n \) is singular, a regularized version of the above problem is solved by using \( \mathbf{S}_n = \mathbf{S}_n + \epsilon \mathbf{I} \), where \( \mathbf{I} \in \mathbb{R}^{D \times D} \) is the identity matrix and \( \epsilon \) is a small positive value.

A Spectral Regression \([12]\) based solution of (4) has been proposed in \([4, 5]\). Let us denote by \( \mathbf{w} \) an eigenvector of the generalized eigen-analysis problem \( \mathbf{S}_n \mathbf{w} = \lambda \mathbf{S}_p \mathbf{w} \) with eigenvalue \( \lambda \). By setting \( \mathbf{Xw} = \mathbf{v} \) (\( \mathbf{X} \) being the data matrix), the original eigen-analysis problem can be transformed to the following eigen-analysis problem \( \mathbf{P}_n \mathbf{v} = \lambda \mathbf{P}_p \mathbf{v} \), where \( \mathbf{P}_n = \mathbf{e}_n \mathbf{e}_n^T - \frac{1}{N_n} \mathbf{e}_n \mathbf{e}_n^T \mathbf{e}_n^T + \frac{1}{N_p} \mathbf{e}_n \mathbf{e}_n^T \mathbf{e}_n^T \mathbf{e}_n \) and \( \mathbf{P}_p = (1 - \frac{1}{N_p} + \frac{1}{N_n}) \mathbf{e}_p \mathbf{e}_p^T \). Here \( \mathbf{e}_p \in \mathbb{R}^N \) and \( \mathbf{e}_n \in \mathbb{R}^N \) are the positive and negative class indicator vectors. That is, \( e_{p,i} = 1 \) if \( \mathbf{x}_i \in \mathcal{S}_p \) and \( e_{p,i} = 0 \) for \( \mathbf{x}_i \in \mathcal{S}_n \), and \( e_{n,i} = 1 - e_{p,i} \). Based on the above, the projection matrix \( \mathbf{W} \) optimizing (4) is obtained by applying a two-step process:

- Solution of the eigen-analysis problem \( \mathbf{P}_n \mathbf{v} = \lambda \mathbf{P}_p \mathbf{v} \), leading to the determination of a matrix \( \mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_d] \), where \( \mathbf{v}_i \) is the eigenvector corresponding to the \( i \)-th largest eigenvalue.
- Calculation of \( \mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_d] \), where:

\[
\mathbf{Xw}_i = \mathbf{v}_i, \quad i = 1, \ldots, d. \quad (7)
\]

It has been shown in \([13]\) that by exploiting the structure of \( \mathbf{P}_p \) and \( \mathbf{P}_n \), their generalized eigenvectors \( \mathbf{v} \) can be directly obtained using the labeling information of the training vectors. However, in that case the order of the eigenvectors is not related to their discrimination ability. Finally, it has been shown in \([14]\) that the class-specific discriminant analysis problem in (1) can be casted as a low-rank regression problem in which the target vectors are the same as those defined in \([13]\), providing a new proof of the equivalence between class-specific discriminant analysis and class specific spectral regression.

After determining the data projection matrix \( \mathbf{W} \), the training vectors \( \mathbf{x}_i, \quad i = 1, \ldots, N \) are mapped to the discriminant subspace \( \mathbf{z}_i = \mathbf{W}^T \mathbf{x}_i \). When a classification problem is considered, a classifier is trained using \( \mathbf{z}_i \). For example, a linear SVM is used in \([5]\) that is trained on vectors \( \mathbf{d}_i = [\mathbf{z}_i - \mu] \), where \( \mu \) is the mean vector of the positive samples in the discriminant subspace and the absolute value operator is applied element-wise on the resulting vector. Due to the need of training an additional classifier, class-specific discriminant analysis models are usually employed in class-specific ranking settings, where test vectors \( \mathbf{x}_j \) are mapped to the discriminant subspace \( \mathbf{z}_j = \mathbf{W}^T \mathbf{x}_j \) and are subsequently ordered based on their distance w.r.t. the positive class mean \( \| \mathbf{z}_j - \mu \|_2 \).

### III. Probabilistic Class-Specific Learning

In this section, we follow a probabilistic approach for defining a class-specific discrimination criterion that is able to encode subclass information of the negative class. We call the proposed method Probabilistic Class-Specific Discriminant Analysis (PCSDA). PCDA assumes that exists a data generation process for the positive class following the distribution:

\[
P(\mathbf{x}) \sim N(\cdot | \mathbf{m}, \Phi_p), \quad (8)
\]

where \( \mathbf{m} \) is the mean vector of the positive class and \( \Phi_p \) is the covariance of the underlying data generation process. A (multi-modal) negative class is formed by subclasses the representations of which are drawn from the following distribution:

\[
P(\mathbf{y}) \sim N(\cdot | \mathbf{m}, \Phi_n). \quad (9)
\]

Here \( \mathbf{y} \) denotes the representations of the negative subclasses and \( \Phi_n \) is the covariance of the negative subclass generation process. Samples of the negative subclasses are drawn from the following distribution:

\[
P(\mathbf{x}) = \int N(\mathbf{y} | \mathbf{m}, \Phi_n) N(\mathbf{x} | \mathbf{y}, \Phi_w) \, d\mathbf{y}, \quad (10)
\]

where \( \Phi_w \) is the covariance of the underlying data generation process for each subclass of the negative class. As we will show later, a special case of the above definition for
the negative class is when each subclass is formed by one sample, leading to the class discrimination criterion used by the standard CSDA and its variants.

Given a set of positive samples \( S_p = \{x_1, \ldots, x_N \} \) the probability of correct assignment under our model (under the assumption that the data are i.i.d.) is equal to:

\[
P(S_p) = \prod_{i=1}^{N_p} P(x_i)
\]

(11)

Let us assume that negative class is formed by \( K \) subclasses, i.e., \( S_n = \{S_1, \ldots, S_K\} \), each having a cardinality of \( M = N_n/K \). We will show how to relax this assumption in the next subsection. Then the probability of assigning each of the negative samples to the corresponding subclass is equal to:

\[
P(S_k) = \int N(y|m, \Phi_n) \prod_{x_i \in S_k} P(x_i|y, \Phi_w) dy.
\]

(12)

By centering the training samples to the positive class mean (this can always be done by setting \( x_i - m \rightarrow x_i \)) we have:

\[
P(S_p) = \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{|\Phi_p|^{\frac{M}{2}}} \exp \left( -\frac{1}{2} Tr(\Phi_p^{-1}S_p) \right)
\]

and

\[
P(S_k) = \frac{1}{M} \frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{|\Phi_w|^{\frac{M}{2}}} \prod_{x_i \in S_k} \exp \left( -\frac{1}{2} Tr(\Phi_w^{-1}S_k) \right)
\]

(13)

In the above, \( S_p \) is the scatter matrix of the positive class, i.e. \( S_p = \sum_{x_i \in S_p} x_i x_i^T \), \( S_w^{(k)} \) is the within-subclass scatter matrix of the \( k \)-th subclass of the negative class, i.e. \( S_w^{(k)} = \sum_{x_i \in S_k} (x_i - \bar{x}_k)(x_i - \bar{x}_k)^T \) and \( S_n^{(k)} \) is the scatter matrix of the \( k \)-th subclass w.r.t. the positive class, i.e. \( S_n^{(k)} = \bar{x}_k \bar{x}_k^T \), where \( \bar{x}_k \) is the mean vector of the \( k \)-th subclass of the negative class, i.e. \( \bar{x}_k = \frac{1}{M} \sum_{x_i \in S_k} x_i \). Since the assignment of the negative samples to subclasses is not provided by the labels used during training, we define them by applying a clustering method (e.g. \( K \)-Means) on the negative class vectors.

### A. Training phase

The parameters of the proposed PCSDA are the covariance matrices \( \Phi_p \), \( \Phi_n \) and \( \Phi_w \) defining the data generation processes for the positive and negative classes. These parameters are estimated by maximizing the (log) probability of correct assignment of the vectors \( x_i \), \( i = 1, \ldots, N \):

\[
L = \ln P(x_i \in S_p) + \ln P(x_i \in S_n),
\]

(15)

where

\[
\ln P(x_i \in S_p) = C_1 - \frac{N_p}{2} \ln |\Phi_p| - \frac{1}{2} Tr(\Phi_p^{-1}S_p)
\]

(16) and

\[
\ln P(x_i \in S_n) = \sum_{k=1}^{K} \ln P(S_k) = C_2 - \frac{N_n}{2} \ln |\Phi_n| - \frac{K}{2} \ln |\Phi_n + \frac{1}{M} \Phi_w| - \frac{1}{2} Tr(\Phi_n^{-1}S_n) - \frac{1}{2} Tr((\Phi_n + \frac{1}{M} \Phi_w)^{-1}S_n).
\]

(17)

\( S_n \) and \( S_w \) are the total scatter matrices of the negative subclasses, i.e., \( S_w = \sum_{k=1}^{K} S_w^{(k)} \) and \( S_n = \sum_{k=1}^{K} S_n^{(k)} \). By substituting (16) and (17) in (15) the optimization function takes the form:

\[
L = C_3 - \frac{N_p}{2} \ln |\Phi_p| - \frac{1}{2} Tr(\Phi_p^{-1}S_p) - \frac{N_n}{2} \ln |\Phi_n| - \frac{K}{2} \ln |\Phi_n + \frac{1}{M} \Phi_w| - \frac{1}{2} Tr((\Phi_n + \frac{1}{M} \Phi_w)^{-1}S_n).
\]

(18)

The saddle points of \( L \) with respect to \( \Phi_p, \Phi_n \) and \( \Phi_w \) lead to:

\[
\frac{\partial L}{\partial \Phi_p} = 0 \Rightarrow \Phi_p = \frac{1}{N_p} S_p
\]

(19)

\[
\frac{\partial L}{\partial \Phi_n} = 0 \Rightarrow \Phi_n + \frac{1}{M} \Phi_w = \frac{1}{K} S_n
\]

(20)

\[
\frac{\partial L}{\partial \Phi_w} = 0 \Rightarrow \Phi_w = \frac{1}{N_n - K} S_w.
\]

(21)

Note that when \( N_n = K \) (21) takes an indeterminate form (i.e. 0/0). However, in this case each negative subclass is formed by one sample and by definition \( \Phi_w = 0 \). This is discussed more in subsection III-C. Combining (20) and (21) we get:

\[
\Phi_n = \frac{1}{K} S_n - \frac{1}{M(N_n - K)} S_w.
\]

(22)

By combining (21) and (22) we can define the total scatter matrix for the negative class:

\[
\Phi_t = \Phi_n + \Phi_w = \frac{1}{K} S_n + \frac{1}{N_n} S_w = S_t.
\]

(23)

Let us denote by \( Q \in \mathbb{R}^{D \times D} \) the eigenvectors of the generalized eigen-analysis problem:

\[
\Phi_t Q = \lambda Q
\]

(24)

The columns of \( Q \) diagonalize both \( \Phi_p \) and \( \Phi_t \), i.e. \( Q^T \Phi_t Q = I \) and \( Q^T \Phi_p Q = \Delta_t \), where \( \Delta_t \in \mathbb{R}^{D \times D} \) is diagonal matrix. Setting \( V = Q^T \), we have:

\[
\Phi_p = V V^T \quad \text{and} \quad \Phi_t = V \Delta_t V^T.
\]

(25)

Let \( W \in \mathbb{R}^{D \times D} \) be the solution of the eigen-analysis problem \( \left( \frac{N_p}{N_n} S_n + \frac{N_n}{N_n} S_w \right) w = \lambda S_p w \). From (19) and (25):

\[
V = W^{-T} \left( \frac{1}{N_p} \Lambda_p \right)^{\frac{1}{2}},
\]

(26)

where \( \Lambda_p = W^T S_p W \) is a diagonal matrix scaling the columns of \( W^{-T} \) to calculate \( V \). Combining (24), (25) and (26):

\[
\Delta_t = N_p \Lambda_t \Lambda_p^{-1},
\]

(27)
where \( \Lambda_c = W^T S_c W \). Thus, \( V \) and \( \Delta_c \) can be computed by solving an eigen-analysis problem defined on the input vectors \( x_i, i = 1, \ldots, N \).

In the above we set the assumption that the number of samples forming the negative classes is equal. This assumption can be relaxed following one of the following approaches. After assigning all negative samples to subclasses and calculating the conditional distributions of each subclass, one can sample \( M \) vectors from these distributions. Alternatively, one can calculate the total within-subclass matrix of the negative class \( S_w \) and define the subclass scatter matrices as \( S_i = \frac{1}{K} S_w \). Note that for the calculation of the model’s parameters (Eqs. \( 19 \) and \( 23 \)), only the total scatter matrices \( S_p, S_w \) and \( S_n \) are used.

B. Test phase

After the estimation of the model’s parameters, a new sample represented by the vector \( x^* \) can be evaluated. The posterior probabilities of the positive and negative classes are given by:

\[
P(c_p|x^*) = \frac{p(x^*|c_p)P(c_p)}{p(x^*)} \tag{28}
\]

\[
P(c_n|x^*) = \frac{p(x^*|c_n)P(c_n)}{p(x^*)} \tag{29}
\]

The a priori class probabilities \( P(c_p) \) and \( P(c_n) \) can be calculated by the proportion of the positive and negative samples in the training phase, i.e. \( P(c_p) = N_p/N \) and \( P(c_n) = N_n/N \). Depending on the problem at hand, it may be sometimes preferable to consider equiprobable classes, i.e. \( P(c_p) = P(c_n) = 1/2 \), leading to the maximum likelihood classification case. The class-conditional probabilities of \( x^* \) are given by:

\[
p(x^*|c_p) = \frac{1}{(2\pi)^{D/2} |\Phi_p|^1/2} \exp \left( -\frac{1}{2} (x^* - \Phi_p x) \right)
\]

\[
= \frac{1}{(2\pi)^{D/2} |V^T|^1/2} \exp \left( -\frac{1}{2} z^T z \right) \tag{30}
\]

and

\[
p(x^*|c_n) = \frac{1}{(2\pi)^{D/2} |\Phi_n|^1/2} \exp \left( -\frac{1}{2} (x^* - \Phi_n x) \right)
\]

\[
= \frac{1}{(2\pi)^{D/2} |\Sigma|^1/2} \exp \left( -\frac{1}{2} z^T \Sigma^{-1} z \right) \tag{31}
\]

where \( z^* = V^{-1} x^* \). In the case where we want \( z^* \in \mathbb{R}^d, d < D \), we keep the dimensions of \( z^* \) corresponding to the \( d \) largest diagonal elements of \( \Delta_c \).

Combining (28)-(31) the ratio of class posterior probabilities is equal to:

\[
\lambda(x^*) = \frac{P(c_p)}{P(c_n)} \left| \frac{\Delta_c}{\Sigma} \right|^{1/2} \exp \left( -\frac{1}{2} z^T \Delta_c^{-1} z \right) \tag{32}
\]

and can also be used to define a classification rule:

\[
g(x^*) = \ln P(c_p) - \ln P(c_n) + \frac{1}{2} \ln \Delta_c
\]

\[
- \frac{1}{2} z^T \Delta_c^{-1} z + \frac{1}{2} z^T \Sigma^{-1} z \tag{33}
\]

classifying \( x^* \) to the positive class if \( g(x^*) > 0 \) and to the negative class otherwise.

In class-specific ranking settings one can follow the process applied when using the standard CSDA approach. First, test vectors \( x_i^* \) are mapped to the discriminant subspace \( z_i^* = W^T x_i^* \) and \( z_i^* \), \( i = 1, \ldots, N \) are obtained. Then, \( z_i^* \)'s are ordered based on their distance w.r.t. the positive class mean \( d_j = \| z_j^* - \mu \|^2 \).

C. CSDA variants under the probabilistic model

A special case of the PCSDA can be obtained by setting the assumption that each negative sample forms a negative subclass, i.e. \( K = N_n \) and \( M = 1 \). In that case \( \Phi_w = 0 \), \( \Phi_1 = \Phi_n \), the negative samples are drawn from a distribution \( P(x) \sim N(x; \Phi_w) \), and \( W \) is calculated by solving for \( S_w = \lambda S_w \), where \( \lambda = \lambda N_n / N_p \), i.e., we obtain the class discrimination definition of CSDA. The Spectral Regression-based solutions of CSDA in [14, 15, 13] and the low-rank regression solution of [14] use the same class discrimination criterion and, thus, correspond to the same setting of PCSDA. Since the discrimination criterion used in CSDA is a special case of the proposed probabilistic model, all the above-mentioned methods can be extended to perform classification using \( g(\cdot) \) in (33).

D. Non-linear PCSDA

In the above analysis we considered the linear class-specific subspace learning case. In order to non-linearly map \( x_i \in \mathbb{R}^D \) to \( z_i \in \mathbb{R}^d \) traditional kernel-based learning methods perform a non-linear mapping of the input space \( \mathbb{R}^D \) to the feature space \( \mathcal{F} \) using a function \( \phi(\cdot) \), i.e. \( x_i \in \mathbb{R}^D \rightarrow \phi(x_i) \in \mathcal{F} \). Then, linear class-specific projections are defined by using the training data in \( \mathcal{F} \). Since the dimensionality of \( \mathcal{F} \) is arbitrary (virtually infinite), the data representations in \( \mathcal{F} \) cannot be calculated. Traditional kernel-based learning methods address this issue by exploiting the Representer theorem and the non-linear mapping is implicitly performed using the kernel function encoding dot products in the feature space, i.e. \( \kappa(x_i, x_j) = \phi(x_i)^T \phi(x_j) \) [15].

As has been shown in [16] the effective dimensionality of the kernel space \( \mathcal{F} \) is at most equal to \( L = \min(D, N) \) and, thus, an explicit non-linear mapping \( x_i \in \mathbb{R}^D \rightarrow \phi_i \in \mathbb{R}^L \) can be calculated such that \( \kappa(x_i, x_j) = \phi_i^T \phi_j \). This is achieved by using \( \Phi = \Sigma^{1/2} U^T \), where \( U \) and \( \Sigma \) contain the eigenvectors and eigenvalues of the kernel matrix \( K \in \mathbb{R}^{N \times N} \) [17]. Thus, extension of PCSDA to the non-linear (kernel) case can be readily obtained by applying the above-described linear PCSDA on the vectors \( \phi_i, i = 1, \ldots, N \). For the cases where the size of training set is prohibitive for applying kernel-based discriminant learning, the Nyström-based kernel subspace learning method of [18] or nonlinear data mappings based on randomized features, as proposed in [19], can be used. Here we should note that the application of \( K \)-Means using in \( \mathbb{R}^L \) corresponds to the application of \( K \)-Means in the original space \( \mathbb{R}^D \).
### IV. EXPERIMENTS

In this Section we provide experimental results illustrating the performance of the proposed PCSDA in comparison with existing CSDA variants. For each dataset, we formed class-specific ranking problems for each class using the class data as positive samples and the data of the remaining classes as negative samples. In all the experiments the data representations are first mapped to the PCA space preserving 98% of the subspace energy and subsequently non-linearly mapped to the subspace of the kernel space (as discussed in Section III-D). We used the RBF kernel function setting the value of $\sigma$ equal to the mean distance value between the positive training vectors. The discriminant subspace is subsequently obtained by applying one of the competing discriminant methods. The dimensionality of the discriminant subspace is determined by applying five-fold cross-validation on the training data within the range of $[1, 25]$. Finally, performance is measured in the discriminant subspace using the average precision metric.

Table II illustrates the performance over all classes of the MNIST dataset [20]. Since by using the entire training set performance saturates, we formed a reduced problem by using only the first 100 images of each class in the training set and report performance on the entire test set. Table III illustrates the performance over all classes of the JAFFEE dataset [21]. We used the grayscale values of the facial images as a representation. For class-specific ranking problem, we applied five experiments by randomly splitting each class in 70%/30% subsets and using the first subset of each class...
to form the training set and the second ones to form the test set. Table III illustrates the performance over all classes of the 15 scene dataset [22]. We employed the deep features generated by average pooling over spatial dimension of the last convolution layer of VGG network [23] trained on ILSVRC2012 database. For each class-specific ranking problem, we applied five experiments by randomly splitting each class using 70%/30% splits.

In Tables I-III the variants of the proposed method that outperform the corresponding baseline are highlighted for each problem. As can be seen, the performance obtained for the standard CSDA and its probabilistic version PCSDA ($K = N_n$) is very similar. The differences observed are mainly due to the small differences in the ranking of the used projection vectors (Eq. (27)). Moreover, exploiting subclass information of the negative class can be beneficial since performance improvements are observed in several cases. Comparing the CSSR in [14] and its probabilistic version PCSSR, it can be seen that the latter improves the performance considerably. This is due to the use of (27), which ranks all eigenvectors of CSSR according to their class-specific discrimination power.

Finally, in Table IV we compare the performance of CSDA and PCSDA for classification. Since each of the class-specific classification problems is imbalanced, we used the g-mean metric, which is defined as the square-root of the multiplication of the correct classification rate of each class. Classification is performed by using the classification rule $g(\cdot)$ in (33) for PCSDA. For CSDA a linear SVM is used is trained on vectors $d_i = |z_i - \mu|$, similar to [5]. The value of $C$ is optimized in the range $10^{(0 \text{--} 3)}$ jointly with the discriminant subspace dimensionality by applying five-fold cross-validation on the training data. Comparing the performance of the CSDA+SVM classification scheme with that of PCSDA ($K = N_n$) on the MNIST and JAFFE datasets, we can see that the use of an additional classifier trained on the data representations in the discriminant subspace increases classification performance. By allowing the negative class to form subclasses the performance increases for PCSDA and is competitive of that of the CSDA+SVM classification scheme. Overall, the PCSDA model achieves competitive performance without requiring the training of a new classifier in the discriminant subspace.

| TABLE IV | PERFORMANCE ON CLASSIFICATION PROBLEMS. |
|----------|------------------------------------------|
|          | MNIST | 15 scene | JAFFE |
| CSDA     | 0.9531 | 0.9119 (0.0074) | 0.7439 (0.0679) |
| PCSDA ($K = N_n$) | 0.9362 | 0.9144 (0.0125) | 0.7102 (0.1164) |
| PCSDA ($K = 5$) | 0.9368 | 0.9132 (0.0143) | 0.7169 (0.0665) |
| PCSDA ($K = 10$) | 0.9367 | 0.9106 (0.0145) | 0.7427 (0.0762) |
| PCSDA ($K = 20$) | 0.9343 | 0.9130 (0.0156) | 0.7268 (0.0592) |

V. CONCLUSIONS

In this paper we proposed a probabilistic model for class-specific discriminant subspace learning that is able to incorporate subclass information naturally appearing in the negative class in the optimization criterion. The adoption of a probability-based optimization criterion for class-specific discriminant subspace learning leads to a classifier defined on the data representations in the discriminant subspace. We showed that the proposed approach includes as special cases existing class-specific discriminant analysis methods. Experimental results illustrated the performance of the proposed model, in comparison with that of related methods, in both verification and classification problems.

REFERENCES

[1] Y. Kittler, J. Li, and J. Matas, “Face verification using client specific Fisher faces,” The Statistics of Directions, Shapes and Images, pp. 65–66, 2000.
[2] G. Goudelis, S. Zafeiriou, A. Tefas, and I. Pitas, “Class-specific kernel discriminant analysis for face verification,” IEEE Transactions on Information Forensics and Security, vol. 2, no. 3, pp. 570–587, 2007.
[3] S. Zafeiriou, G. Tzimiropoulos, M. Petrou, and T. Stathaki, “Regularized kernel discriminant analysis with a robust kernel for face recognition and verification,” IEEE Transactions on Neural Networks, vol. 23, no. 3, pp. 526–534, 2012.
[4] S. Arashloo and J. Kittler, “Class-specific kernel fusion of multiple descriptors for face verification using multiscale binarized statistical image features,” IEEE Transactions on Information Forensics and Security, vol. 9, no. 12, pp. 2100–2109, 2014.
[5] A. Iosifidis, A. Tefas, and I. Pitas, “Class-specific reference discriminant analysis with application in human behavior analysis,” IEEE Transactions on Human-Machine Systems, vol. 45, no. 3, pp. 315–326, 2015.
[6] A. Iosifidis, M. Gabbouj, and P. Pecki, “Class-specific nonlinear projections using class-specific kernel spaces,” IEEE International Conference on Big Data Science and Engineering, pp. 1–8, 2015.
[7] A. Iosifidis, A. Tefas, and A. Pitas, “Activity based person identification using fuzzy representation and discriminant learning,” IEEE Transactions on Information Forensics and Security, vol. 7, no. 2, pp. 530–542, 2012.
[8] R. Duda, P. Hart, and D. Stork, Pattern Classification, 2nd ed. Wiley-Interscience, 2000.
[9] A. Iosifidis, A. Tefas, and I. Pitas, “On the optimal class representation in Linear Discriminant Analysis,” IEEE Transactions on Neural Networks and Learning Systems, vol. 24, no. 9, pp. 1491–1497, 2013.
[10] A. Iosifidis, A. Tefas, and I. Pitas, “Kernel reference discriminant analysis,” Pattern Recognition Letters, vol. 49, pp. 85–91, 2014.
[11] Y. Jia, F. Nie, and C. Zhang, “Trace ratio problem revisited,” IEEE Transactions on Neural Networks, vol. 20, no. 4, pp. 729–735, 2009.
[12] D. Cai, X. He, and J. Han, “Spectral regression for efficient regularized subspace learning,” International Conference on Computer Vision, 2007.
[13] A. Iosifidis and M. Gabbouj, “Scaling up class-specific kernel discriminant analysis for large-scale face verification,” IEEE Transactions on Information Forensics and Security, vol. 11, no. 11, pp. 2453–2465, 2016.
[14] A. Iosifidis and M. Gabbouj, “Class-specific kernel discriminant analysis revisited: Further analysis and extensions,” IEEE Transactions on Cybernetics, vol. 47, no. 12, pp. 4485–4496, 2017.
[15] B. Scholkopf and A. Smola, Learning with Kernels. MIT Press, 2001.
[16] N. Kwak, “Nonlinear Projection Trick in kernel methods: an alternative to the kernel trick,” IEEE Transactions on Neural Networks and Learning Systems, vol. 24, no. 12, pp. 2113–2119, 2013.
[17] N. Kwak, “Implementing kernel methods incrementally by incremental nonlinear projection trick,” IEEE Transactions on Cybernetics, vol. 47, no. 11, pp. 4003–4009, 2017.
[18] A. Iosifidis and M. Gabbouj, “Nyström-based approximate kernel subspace learning,” Pattern Recognition, vol. 57, pp. 190–197, 2016.
[19] A. Rahimi and B. Recht, “Random features for large-scale kernel machines.” Advances in Neural Information Processing Systems, 2007.
[20] Y. LeCun, L. Bottou, and Y. Bengio, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, pp. 2278–2324, 1998.
[21] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, “Coding facial expressions with gabor wavelets,” IEEE International Conference on Automatic Face and Gesture Recognition, pp. 200–205, 1998.
[22] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” IEEE Conference on Computer Vision and Pattern Recognition, 2006.
[23] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition.” arXiv:1409.1556, 2014.