Gaussian Mixture Models via Direct Nonparametric Discriminant Analysis for Classification of High-Dimensional Data

Xianglei Xing, Sidan Du, Yu Zhou, Yao Yu, Yang Li
School of Electronic Science and Engineering, Nanjing University of China
Email: xlxing@ese.nju.edu.cn, coff128@nju.edu.cn

Abstract

The Gaussian mixture model (GMM) is a powerful tool for data clustering and pattern classification. GMM, which inherits the intrinsic strengths of generative models, can approximate arbitrary probability distributions. However, it suffers from the small size samples problem, particularly in the very high-dimensional feature space. GMM, like other generative models, is also inferior to discriminative models on generalization performance. In this paper, we present a regularized GMM via direct nonparametric discriminant analysis method (DNDA-GMM) for classifying high-dimensional data. The proposed direct nonparametric discriminant analysis (DNDA) algorithm accepts high dimensional data as input directly, and optimizes the classification boundary effectively just like discriminative approaches. DNDA also achieves dimension reduction so as to alleviate the issues within GMM in very high dimensions. Thus, the proposed DNDA-GMM classifier takes advantage of both the flexibility provided by generative models and the classification performance increases provided by discriminative approaches. Experimental results on real world data sets demonstrate the effectiveness of the proposed approach.

1. Introduction

GMM is widely used in pattern recognition, machine learning, and statistical analysis. It has enjoyed wide spread applications in engineering including image segmentation [1], speech emotion recognition [2] and micro-array gene expression data analysis [3]. GMM's success comes from several intrinsic strengths of the generative modeling approach to classification as well as the power of mixture modeling as a density estimation method for multivariate data [4].

Generative modeling methods and discriminative approaches are two major paradigms for pattern classification. One way to view discriminative approaches is in terms of dimensionality reduction. Although strictly speaking, nonparametric discriminant analysis is not a discriminant but rather a dimensionality reduction technique. However, the projected data can subsequently be used to construct a discriminant by choosing an optimal threshold. For example, we can model the class-conditional densities using Gaussian distributions and obtain the optimal threshold by minimizing the misclassification rate. Some justification for the Gaussian assumption comes from the central limit theorem by noting that $y = W^T x$ is the sum of a set of random variables [5].

Practical issues arise when applying EM algorithm to a mixture of Gaussian in the very high-dimensional feature space, especially there are far more feature dimensions than training samples. In the above situation, singular or nearly singular covariance will appear during the iterative procedure in EM. It will break down the convergence of EM and cause numerical breakdown of GMM. Challenging to compute big matrices, to handle singular or nearly singular matrices also exists in the traditional nonparametric discriminant analysis algorithm. In this paper a DNDA-GMM classifier (Gaussian Mixture Models via Direct Nonparametric Discriminant Analysis), which utilizes the relative benefits of both the generative and discriminative learning paradigms, was proposed to solve above problems.

2. Gaussian Mixture Model and Practical Issues in High Dimension

2.1. Gaussian Mixture Model for Classification

Let $\boldsymbol{X} = (\boldsymbol{x}_1,\boldsymbol{x}_2,\ldots,\boldsymbol{x}_n)$ be a set of data points, and the class label of $\boldsymbol{X}$ be $Z \in \{1, 2, \ldots, K\}$. The overall model for a joint distribution of $\boldsymbol{X}$ and Z under a Gaussian mixture is:

$$P(\boldsymbol{X} = \boldsymbol{x}, Z = k) = a_k \sum_{r=1}^{R_k} \pi_{kr} \phi(\boldsymbol{x}|\boldsymbol{\mu}_{kr}, \boldsymbol{\Sigma}_{kr}) \quad (1)$$

Where a_k is the prior probability of class k, satisfying $0 \leq a_k \leq 1$ and $\sum_{k=1}^{K} a_k = 1$. The ML estimation of a_k is the proportion of training samples in class k. Each class is modeled by a mixture of Gaussian. For class k, the within-class density is: $P_k(\boldsymbol{x}) = \sum_{r=1}^{R_k} \pi_{kr} \phi(\boldsymbol{x}|\boldsymbol{\mu}_{kr}, \boldsymbol{\Sigma}_{kr})$ where π_{kr} is the mixing parameters of the rth component in class k, such that $0 \leq \pi_{kr} \leq 1, \sum_{r=1}^{R_k} \pi_{kr} = 1$. $\phi(\cdot)$ is a Gaussian density function in the form

$$\phi(\boldsymbol{x}|\boldsymbol{\mu}_{kr}, \boldsymbol{\Sigma}_{kr}) = \frac{e^{-(1/2)(\boldsymbol{x}-\boldsymbol{\mu}_{kr})^T \boldsymbol{\Sigma}_{kr}^{-1} (\boldsymbol{x}-\boldsymbol{\mu}_{kr})}}{(2\pi)^{d/2} |\boldsymbol{\Sigma}_{kr}|^{1/2}} \quad (2)$$

parameterized by mean vector $\boldsymbol{\mu}_{kr}$ for component r of class k, and corresponding covariance matrix $\boldsymbol{\Sigma}_{kr}$. R_k is the number of mixture components in class k. The total number of mixture components for all classes is denoted by $M = \sum_{k=1}^{K} R_k$. Roughly speaking, we estimate a mixture of normals by EM for each individual class. EM consists of two steps: an expectation step (E-step) and a maximization step (M-step). The expectation step is to calculate the probability of the data coming from the current model, and give the current estimates of the observed data and the hidden parameters. The E-step is defined as $Q(\Theta, \Theta^p) = E[\log p(\boldsymbol{X}, Z|\Theta)|\boldsymbol{X}, \Theta^p]$. Here Θ is the model parameters including $\boldsymbol{\pi}$, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. The maximization step is to choose the values of Θ that maximize the probability calculated in E-step. The M-step is defined as $\Theta^{p+1} = argmax_\Theta Q(\Theta, \Theta^p)$. For details about EM, please see [6].

2.2. Practical Issues in High Dimension

Practical issues arise when applying EM to Gaussian Mixture Model in the very high-dimensional feature space.

First of all, severe hurdles will be encountered in the very high-dimensional feature space. In some practical scenarios, such as imagery data and micro-array gene expression data, each sample can easily reach 10000 dimensions or even higher. However, it is practically impossible to estimate such a large covariance matrix. In order to alleviate this problem, diagonal covariance matrices are often used in the Gaussian component. It assumes that the features are independent within each class. The assumption of independence greatly reduces the number of parameters in the model and often results in an effective and interpretable classifier [7]. Another solution is to restrict the models to a low dimensional space and dimensionality reduction technique can be used.

Another problem in the high-dimensional feature space is that it suffers from the small samples size problem and the sample covariance matrix will be singular. A singular covariance matrix cannot be inverted and makes it difficult to compute the probability in Eq.(2), causing numerical breakdown of GMM.

A solution for singular covariance matrix is to introduce a regularization term, which is proposed by Friedman [8]. The covariance matrix $\boldsymbol{\Sigma}_{kr}$ is regularized with a multiple of identity matrix(or global covariance matrix), so as to have $\boldsymbol{\Sigma}_{kr} = \boldsymbol{\Sigma}_{kr} + \gamma \boldsymbol{I}$. The effect of the regularization is to stabilize the smallest eigenvalues by increasing the smaller ones and decreasing the larger eigenvalues.

3. Gaussian Mixture Models via Direct Nonparametric Discriminant Analysis

3.1. Parametric and Nonparametric Discriminant Analysis

Parametric discriminant analysis (PDA) is a dimensionality reduction technique that extends Fisher's linear discriminant analysis (LDA) to multiple classes [9]. In PDA, the data are projected from the original L dimensional space to a $K-1$ dimensional subspace through a optimal linear transformation matrix, such that ratio of the determinant of between-class matrix to that of the within-class matrix is maximized. The optimal projection matrix can be obtained by solving a generalized eigenvalue problem.

There are two disadvantages in PDA. First, the rank of the between class matrix is at most $K-1$, so the number of the final PDA feature is no more than $K-1$. However, it is often insufficient to separate the classes well with only $K-1$ features, especially severe if $K \ll L$. Second, the boundary structure of classes is not taken into account for computing between-class scatter matrix, which has been shown to be essential in classification.

Nonparametric discriminant analysis has been proposed to solve the aforementioned problems [10][11]. Nonparametric between-class scatter matrix and within-class scatter matrix are defined as:

$$S_b^N = \sum_{i=1}^{K} \sum_{j=1;j\neq i}^{K} \sum_{l=1}^{N_i} w(i,j,p,l)(x_l^i - NN_p(x_l^i, j)) \\ (x_l^i - NN_p(x_l^i), j)^T \quad (3)$$

$$S_w^N = \sum_{i=1}^{K}\sum_{l=1}^{N_i}(x_l^i - NN_p(x_l^i, i))(x_l^i - NN_p(x_l^i, i))^T \tag{4}$$

where x_l^i denotes the lth samples from class i, $NN_p(x_l^i, j)$ is the pth nearest neighbor from class j to the face vector x_l^i. The weighting function $w(i, j, p, l)$ is defined as: $w(i, j, p, l) =$

$$\frac{min\{d^\alpha(x_l^i, NN_p(x_l^i, i)), d^\alpha(x_l^i, NN_p(x_l^i), j)\}}{d^\alpha(x_l^i, NN_p(x_l^i, i)) + d^\alpha(x_l^i, NN_p(x_l^i), j)} \tag{5}$$

where α is a control parameter between zero and infinity, and $d(x_l^i, NN_p(x_l^i, i))$ is the Euclidean distance between two vectors. The weighting function has the property that near the classification boundary it takes on values close to 0.5 and drops off to zeros if the samples are far away from the classification boundary. This weighting function is used to emphasize the boundary information. The optimal transformation matrix (W_N) is defined as:

$$W_N = argmax\frac{|W_N^T S_b^N W_N|}{|W_N^T S_w^N W_N|} \tag{6}$$

3.2. Direct Nonparametric Discriminant Analysis

The traditional NDA algorithm has several difficulties for task with very high dimensional data. First, the scatter matrices will be always singular. Second, it is challenging to compute these big matrices such as computing eigenvalues. A relevant method for linear discriminant analysis developed in similar situations is the D-LDA [12], where it accepts high dimensional data as input, and optimizes Eq.(6) directly, without any feature extraction or dimensionality reduction steps. Motivated by the success of [12], a DNDA algorithm is developed here for nonparametric discriminant analysis with high dimensional data set.

The trick presented by Turk and Pentland [13] for EigenFace is employed. The nonparametric scatter matrices can be represented in a convenient way that both affords eigen-analysis and saves memory.

$$S_b^N = \Phi_b\Phi_b^T \tag{7}$$

$$\begin{aligned}\Phi_b =&[\sqrt{w_{121}}(x_1^1 - NN_p(x_1^1, 2)), \cdots\\ &\sqrt{w_{211}}(x_1^2 - NN_p(x_1^2, 1)), \cdots\\ &\sqrt{w_{c11}}(x_1^c - NN_p(x_1^c, 1)), \cdots\\ &\cdots, \sqrt{w_{c(c-1)N_c}}(x_{N_c}^c - NN_p(x_{N_c}^c, c-1))]\end{aligned} \tag{8}$$

where $w(i, j, p, l)$ is brief written down for w_{ijl}. S_b^N is a $L \times L$ matrix. Φ_b is a $L \times (K-1)(N_1 + N_2 + \cdots + N_K) = L \times (K-1)N$ matrix (N is the number of training samples). Thus, we need only to store a $L \times (K-1)N$ matrix instead of storing a $L \times L$ matrix. We need only perform the eigen-analysis on a $(K-1)N \times (K-1)N$ matrix rather than a large matrix which is $L \times L$. It can be shown by the following lemma.

***Lemma** 1:* The eigen-analysis of a $L \times L$ matrix $S = \Phi\Phi^T$ with eigenvalue λ_i and eigenvector μ_i can be simplified by evaluating a $m \times m$ matrix $T = \Phi^T\Phi$, where $m < L$, with eigenvalues λ_i and eigenvector ν_i. The two matrices have the same m largest eigenvalues and the corresponding eigenvectors satisfy $\mu_i \propto \Phi\nu_i$ with a constraint such that $||\mu_i|| = 1$.

Assuming that $\mathcal{A}$ and $\mathcal{B}$ represent the non-null space of S_b^N and S_w^N, while $\overline{\mathcal{A}}$ and $\overline{\mathcal{B}}$ are the null space respectively, the optimal discriminant subspace derived from DNDA is the intersection space $\mathcal{A}\bigcap\overline{\mathcal{B}}$. The key idea of DNDA algorithm is to remove the null space of S_b^N, which contains no significant information, and reserve the null space of S_w^N, which contains the most discriminative information. The DNDA algorithm is outlined below.

1) Compute the eigenvectors of $\Phi_b^T\Phi_b$ with nonzero eigenvalues: $V_m = [\nu_1, \cdots, \nu_m]$, where Φ_b is from Eq. (8) and $m \leq (K-1)N$.
2) Diagonalize S_b^N. The first m most significant eigenvectors and the corresponding eigenvalues of S_b^N are $U = \Phi_b V_m$ and $\Lambda_b = V_m^T S_b^N V_m$.
3) Let $Z = U\Lambda_b^{-1/2}$. Diagonalize $Z^T S_w^N Z$ and calculate the eigenvectors P and eigenvalues Λ_w of $Z^T S_w^N Z$. Sort the diagonal elements of Λ_w, and discard some eigenvectors in P with the largest eigenvalues. Let P' and Λ'_w denotes the D selected eigenvectors and eigenvalues($D \leq m$).
4) The projection matrix $W_N = ZP'\Lambda_w'^{-1/2}$ maps the original high dimensional space to a D-dimensional subspace and will be used in the classification stage. The final W_N is defined as:

$$W_N = \Phi_b V_m (V_m^T S_b^N V_m)^{-1/2} P' \Lambda_w'^{-1/2} \tag{9}$$

3.3. The DNDA-GMM Classifier

Details of the DNDA-GMM classifier are described in this section. The DNDA-GMM classifier is a GMM minimum error rate classifier formulating in the DNDA subspace. Now, let $\{x_i, k_i\}$ be a set of N training samples, where $i = 1, \ldots, N$, $x_i \in \mathbb{R}^L$, $k_i \in \{1, \ldots, K\}$. Firstly, project x_i to the DNDA subspace where the most discriminatory features are preserved: $y_i = W_N^T x_i$, where $y_i \in \mathbb{R}^D$ with $D \leq (K-1)N$, and

W_N was defined in Eq. (9). In practice, the optimal subspace dimension D can be chosen based on the performance of the model on validation data, or by cross-validation.

Secondly, train a GMM for each class k in the DNDA subspace: $P_k(x_i|W_N) = P_k(y_i), k \in 1, \cdots, K$, where $P_k(y_i) = \sum_{r=1}^{R_k} \pi_{kr}\phi(y_i|\mu_{kr}, \Sigma_{kr})$. The EM algorithm is used to estimate the model parameters. In the E-step, collect samples in each class k and compute the posterior probabilities of all the R_k components:

$$p_{i,r} = \frac{\pi_{kr}\phi(y_i|\mu_{kr}, \Sigma_{kr})}{\sum_{r'=1}^{R_k} \pi_{kr'}\phi(y_i|\mu_{kr}, \Sigma_{kr})} \tag{10}$$

In the M-step, compute the weighted ML estimation for all the parameters:

$$\pi_{kr} = \frac{\sum_{i=1}^{n} I(z_i = k)p_{i,r}}{\sum_{i=1}^{n} I(z_i = k)} \tag{11}$$

$$\mu_{kr} = \frac{\sum_{i=1}^{n} y_i I(z_i = k)p_{i,r}}{\sum_{i=1}^{n} I(z_i = k)p_{i,r}} \tag{12}$$

$$\Sigma_{kr} = \frac{\sum_{i=1}^{n} I(z_i = k)p_{i,r}(y_i - \mu_{kr})(y_i - \mu_{kr})^T}{\sum_{i=1}^{n} I(z_i = k)p_{i,r}} \tag{13}$$

To initialize the estimate algorithm, R_k, the number of components in a class, is determined by its corresponding proportion in the whole training samples. Within class k, a k-means clustering model, with multiple random starts, is fitted to the data. The initial posterior probabilities $p_{i,r}$ is calculated by the inverse of the Euclidean distance between a sample y_i and cluster center c_r derived from k-means clustering, $p_{i,r} = \frac{1}{||y_i - c_r||}$. The probabilities are then normalized across clusters, so that $\sum_{r=1}^{R_k} p_{i,r} = 1$. If any mixture component happens to be empty according to the k-means clustering, we use global mean to replace that empty clustering mean. Then an M-step is applied to obtain the initial parameters with the initial posterior probabilities given above.

Regularized covariance is used to prevent the covariance from singular during the estimation.

$$\Sigma_{kr} = \lambda\Sigma_{kr} + (1 - \lambda)\Sigma_{global} \tag{14}$$

where Σ_{global} is the global covariance. The optimal regularized parameter $\lambda \in [0, 1)$ will be determined by cross validation in the step of model selection.

Finally, form the minimum error rate classifier and accomplish the classification task.

$$H(y) = \underset{k \in \{1, \cdots, K\}}{\operatorname{argmax}} P_k(y)a_k \tag{15}$$

where a_k is the prior probability of class k.

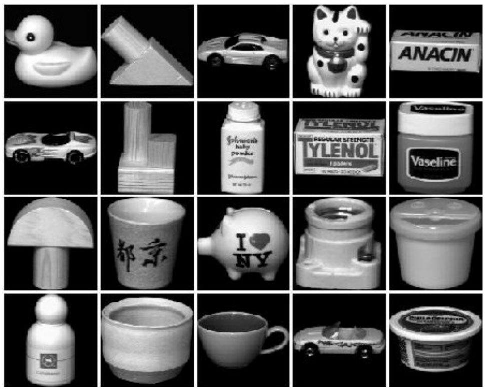

Figure 1. Sample images from COIL20 data set.

4. Experimental Results

In this section, experiments are performed, based on two real world data sets from moderate high to very high dimensions, to show the effectiveness of our proposed algorithm. The first one is Columbia Object Image Library (COIL-20) [14]. COIL-20 is a database of gray-scale images of 20 objects. Images of the objects were taken at pose intervals of 5 degrees and correspond to 72 images per object. The size of each image is 32×32 pixels, with 256 grey levels per pixel. Each image is represented by a 1024-dimensional feature vector. Some sample images are shown in Fig. 1. The first 3 class images are used in the experiment. The second one is the ARCENE data set [15] which is one of five data sets used in the NIPS 2003 feature selection challenge. ARCENE's task is to distinguish cancer versus normal patterns from mass-spectrometric data. Each sample is represented by a 10000 dimensional feature vector. The order of the features and patterns is randomized. There are 200 samples divided into two classes: 88 positive examples and 112 negative examples.

4.1. Classification Results and Discussion

To demonstrate how our algorithm improves the performance of data classification, we compared the following four methods:

- Diagonal Gaussian Mixture Model (Diag-GMM).
- Gaussian Mixture Model with Principal Component Analysis (PCA-GMM).
- Gaussian Mixture Model with Linear Discriminant Analysis (LDA-GMM).
- Gaussian Mixture Model with Direct Nonparametric Discriminant Analysis (DNDA-GMM).

Figure 2 shows the results using the COIL-20 data set. It shows the overall classification error rate of

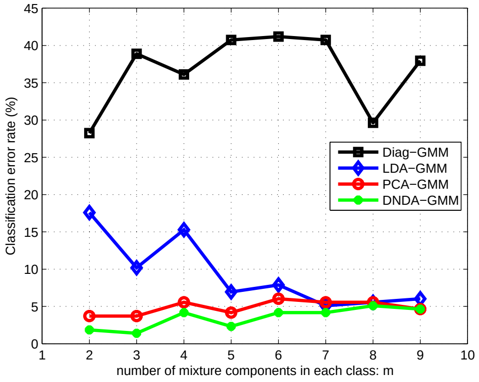

Figure 2. The overall classification error rate of the four classifiers versus the number of mixture components in each class, which ranges from 2 to 9, with $\lambda = 0.95$ and $D = 100$ for COIL-20.

the four classifiers versus the number of each class's mixture components m which ranges from 2 to 9. For PCA-GMM, LDA-GMM and DNDA-GMM, there are two important parameters, that is, regularization parameter λ and reduced subspace dimension D. We will discuss the effect on the selection of different model parameters in the next subsection. The results, in Figure 2, were get with $\lambda = 0.95$ and $D = 100$. For LDA-GMM, the reduced subspace dimension is at most $K - 1$.

As can be seen from Figure 2, DNDA-GMM performs the best of all. PCA-GMM performs the second best, LDA-GMM performs the third and Diag-GMM performs the worst. Diag-GMM assumes that the features are independent within each class. However, features will rarely be independent within a class in practice. The PCA-GMM method, which uses PCA for dimensionality reduction, yields projection directions that maximize the total scatter across all classes. The LDA-GMM method, which uses LDA for dimensionality reduction, optimizes the low-dimensional representation of the objects with focus on the most discrimination between classes and increases the classification performance to a certain degree. However, important discriminatory information may be lost during dimensionality reduction. This problem becomes especially severe when the class number K is much less than the data dimension D. Thus, PCA may outperform LDA when the number of training samples per class is small. As we can see, DNDA-GMM yields a smaller classification error rates than the above. The DNDA-GMM achieves its minimum error rate 1.39% when $m = 3$.

Table 1 shows the experimental results using the ARCENE data set. Five-fold cross validation was used to compute the classification error rate. The overall classification error rate of the three classifiers versus the total number of mixture components M which range from 2 to 18 is shown in the table. LDA-GMM is not contained, since there is only two classes in the ARCENE data set. The results, in Table 1, were get with $\lambda = 0.95$ and $D = 98$. The minimum error rate is in bold font in each column. As Table 1 shows, the smallest error rate is always achieved by the DNDA-GMM, for each column, when the number of mixture components is fixed. It shows the advantage of the DNDA-GMM method.

4.2. Model Selection

Model selection is an important problem in many learning problems. Different choices of the parameters may bring down the learning performance in some situations. There are two essential parameters in our DNDA-GMM algorithm: regularization parameter λ and the subspace dimension D. Data driven method, such as cross validation and grid search, can be used to find the pair of λ and D that gives the lowest error rate. Table 2 provides the average classification error rate(weighted average of error rate with mixture components from 1 to 9) of DNDA-GMM with some different values of λ and D using the ARCENE data set. As we can see from Table 2, firstly, for each column, when the regularization parameter λ is fixed, the higher subspace dimension D is chosen, the smaller classification error rate is achieved. From section 3.2, we find that more information of between class can be preserved by choosing a higher value of D. Secondly, for each row, when the subspace dimension D is fixed, the bigger regularization parameter λ is chosen, the smaller classification error rate is achieved. On the other hand, regularization parameter λ must be smaller than 1 to prevent the covariance from singular.

5. Conclusion

In this research, we have introduced a novel method for classifying high-dimensional data. The proposed DNDA-GMM classifier takes the advantages of both the generative methods and the discriminative approaches. A direct nonparametric discriminant analysis algorithm was proposed for high-dimensional data. DNDA is effective in the utilization of the classification boundary information just like discriminative approaches. In addition to the advantages of optimizing the classification boundary, DNDA mainly accomplishes the dimensions reduction

Table 1. The overall classification error rate of the three classifiers versus the total number of mixture components M, which range from 2 to 18, with $\lambda = 0.95$ and $D = 98$ for the ARCENE data.

Error rate (%)	M = 2	M = 4	M = 6	M = 8	M = 10	M = 12	M = 14	M = 16	M = 18
DiagGMM	34	35.5	17.5	15	14	14.5	15.5	14	15
PCAGMM	23	14.5	12	12	12	12	12.5	11.5	13.5
DNDAGMM	**19.5**	**11**	**10.5**	**9.5**	**10.5**	**11**	**10**	**9.5**	**10**

Table 2. The average classification error rate of DNDA-GMM with parameter λ varying from 0.1 to 0.99 and D varying from 10 to 98 using the ARCENE data set.

Average Error Rate (%)	λ = 0.99	λ = 0.95	λ = 0.9	λ = 0.7	λ = 0.5	λ = 0.3	λ = 0.1
D = 98	**12.56**	14.78	14.67	14.67	15.33	16.0	16.0
D = 80	**14.22**	15.33	15.44	15.78	17.22	20.33	23.22
D = 40	**14.78**	15.22	20.89	35.11	40.56	41.56	42.33
D = 20	**23.67**	35.0	40.33	43.33	44.0	44.0	44.0
D = 10	**34.56**	39.33	42.78	43.78	44.0	44.0	44.0

and alleviates the issues within GMM in very high dimensions. Regularized covariance matrix technique was used in this paper to overcome the problem of singular or nearly singular covariance during the iterative procedure in EM.

Experiments conducted on two real world data sets show that the DNDA-GMM classifier often outperforms the diagonal Gaussian mixture model. The experimental results also show that our method is superior to PCA-GMM and LDA-GMM approaches in terms of classification accuracy and stability.

References

[1] Martinez and J. M. Sotoca, “A semi-supervised gaussian mixture model for image segmentation,” in *Proc. ICPR*, Castellon, Spain, 2010, pp. 2941–2944.

[2] H. Tang and T. Huang, “Boosting gaussian mixture models via discriminant analysis,” in *Proc. ICPR*, Urbana, USA, 2008, pp. 1–4.

[3] M. Qiao and J. Li, “Two-way gaussian mixture models for high dimensional classification,” *Statistical Analysis and Data Mining*, vol. 3, no. 4, pp. 259–271, 2010.

[4] C. Fraley and A. E. Raftery, “Model-based clustering, discriminant analysis, and density estimation,” *Journal od the American Statistical Association*, vol. 97, pp. 611–631, 2002.

[5] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm,” *Journal Royal Statistics Society*, vol. 39, no. 1, pp. 1–21, 1977.

[7] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning*. New York: Springer-Verlag, 2009.

[8] J. H. Friedman, “Regularized discriminant analysis,” *J. Amer. Statist. Assoc.*, vol. 84, no. 405, pp. 165–175, 1989.

[9] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2000.

[10] K. Fukunaga, *Statistical Pattern Recognition*. Academic Press, 1990.

[11] Z. F. Li, D. H. Lin, and X. O. Tang, “Nonparametric discriminant analysis for face recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 755–761, 2009.

[12] H.Yu and J. Yang, “A direct lda algorithm for high-dimensional data with application to face recognition,” *Pattern Recognition*, vol. 34, no. 11, pp. 2067–2070, 2001.

[13] M. Turk and A. Pentland, “Eigenfaces for recognition,” *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 72–86, 1991.

[14] S. A. Nene, S. K. Nayar, and H. Murase. (1996, Feb.) Columbia university image library. [Online]. Available: http://www1.cs.columbia.edu/CAVE/software/softlib/

[15] I. Guyon, S. R. Gunn, A. B. Hur, and G. Dror, “UCI machine learning repository,” 2004. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Arcene