A New Two-layer Mixture of Factor Analyzers with Joint Factor Loading Model for the Classification of Small Dataset Problems

Xi Yang[a], Kaizhu Huang[a,*], Rui Zhang[b], John Y. Goulermas[c], Amir Hussain[d]

[a]*Department of Electrical and Electronic Engineering, Xi'an Jiaotong-Liverpool University, SIP, Suzhou, 215123, P.R.China*
[b]*Department of Mathematical Sciences, Xi'an Jiaotong-Liverpool University, SIP, Suzhou, 215123, P.R.China*
[c]*Department of Computer Science, Ashton Building, University of Liverpool, Liverpool, L69 3BX, UK*
[d]*Division of Computing Science and Maths, School of Natural Sciences, University of Stirling, Stirling FK9 4LA, UK*

Abstract

Dimensionality Reduction (DR) is a fundamental topic of pattern classification and machine learning. For classification tasks, DR is typically employed as a pre-processing step, succeeded by an independent classifier training stage. However, such as independent operation of the two stages often limits the final classification performance notably, as the generated subspace may not be maximally beneficial or appropriate to the learning task at hand. This problem is further accentuated for high-dimensional data classification in situations of limited number of samples. To address this problem, we develop a novel joint learning model for classification, referred to as two-layer mixture of factor analyzers with joint factor loading (2L-MJFA). Specifically, the model adopts a special two-layer mixture or a mixture of mixtures structure, where each component represents each specific class as a mixture of factor analyzers (MFA). Importantly, all the involved factor analyzers are intentionally designed so that they share the same loading matrix. This, apart from operating as the DR

*Corresponding author

Email addresses: `xi.yang@xjtlu.edu.cn` (Xi Yang), `kaizhu.huang@xjtlu.edu.cn` (Kaizhu Huang), `rui.zhang02@xjtlu.edu.cn` (Rui Zhang), `j.y.goulermas@liverpool.ac.uk` (John Y. Goulermas), `ahu@cs.stir.ac.uk` (Amir Hussain)

matrix, it largely reduces the parameters and makes the proposed algorithm very suitable to small dataset situations. Additionally, we propose a modified expectation maximization algorithm to train the proposed model. A series of simulation experiments demonstrates that what we propose significantly outperforms other state-of-the-art algorithms on various benchmark datasets.

Keywords: `Factor analyzer, Joint learning, Classification, Dimensionality reduction`

2010 MSC: 00-01, 99-00

---

1. Introduction

Dimensionality reduction (DR) is a very important topic of pattern recognition and machine learning that has been studied intensely in the relevant literature. Its objective is the finding of a subspace to effectively reduce the computational time while improving the performance of the learning task [1]. Traditionally, DR is performed as a pre-processing step to remove noise and compact the representation. Subsequently, the reduced features can be fed to various models for accurately learning a classification task. A typical example of this workflow, includes a Gaussian mixture model (GMM) classifier applied after a linear DR method, such as principal component analysis (PCA), linear discriminant analysis (LDA), factor analyzer (FA) [2, 3], or a method from the recently proposed [4, 5, 6, 7]. Besides linear methods, there are other DR techniques that achieve nonlinear projections of the data [8, 9].

While the independent realization of DR and classification can be easily implemented, it may notably diminish the final performance [10, 11] as the two tasks do not necessarily interact with each other, and the optimal subspace obtained by the DR may not be maximally beneficial to the learning task. This is particularly the case for the small sample size (S3) problem [12, 13], where the data patterns are high-dimensional but of low cardinality. In such problems, the subspace derived by the independent DR may even significantly deteriorate the classification performance.

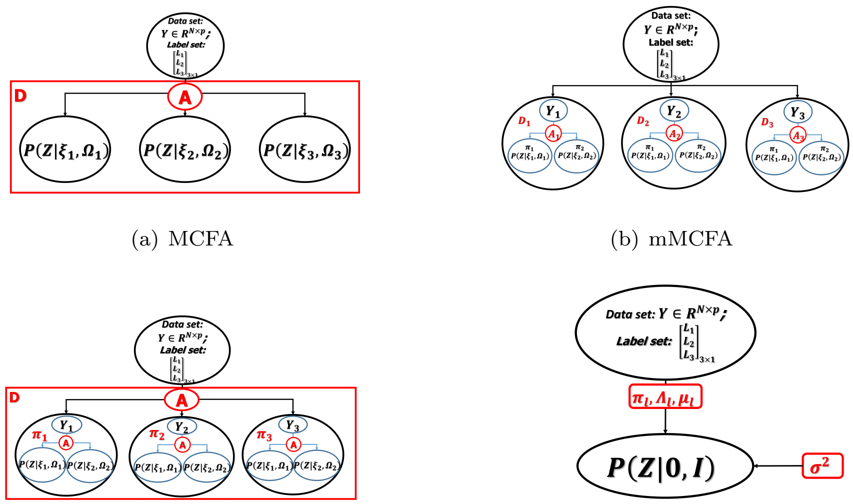
Motivated from the above issues, we propose within an FA framework, a novel model referred to as the two-layer mixture of factor analyzers with joint factor loading (2L-MJFA). This relies upon a mixture of mixtures structure, used to better capture the complex properties of each class and realize efficiently the joint learning requirements. An important characteristic of 2L-MJFA, is that all of its involved latent factors are designed to share the same loading matrix. This has a dual purpose, in the sense that, on one hand it operates as the driving DR structure, and on the other hand it significantly reduces the number of parameters. The latter accelerates training while mitigates the negative effect caused by the limited number of per class samples.

Contrary to the independent approaches, the proposed 2L-MJFA is capable of simultaneously learning the DR matrix as well as the optimal parameters of the classification model. This model is implemented via a GMM for simplicity, but it is straightforward to extend the two-layer mixture approach to the use of other models. Through joint learning, the method achieves efficient DR that not only reduces the computational time for high dimensional data, but more importantly it significantly benefits the final classification stage. Another contribution, is that we also propose a modified expectation-maximization (EM) algorithm that consists of two-layer loops, so that the joint learning is conducted very efficiently. The first layer loop is used to estimate the joint parameters that fit the mixture among different classes, whereas the second one trains the mixture components within each class. The 2L-MJFA is theoretically distinct to other joint learning FA models, such as the FA mixture with common loading (MCFA) [14], the mixture of MCFAs (mMCFA), and the mixture of probabilistic PCA (mPPCA) [15, 16]. Further details about these models are presented in the following section. Our experiments show that the proposed method significantly outperforms these existing methods in seven benchmark datasets.

The rest of this paper is organized as follows. Section 2 briefly reviews related work and emphasizes the differences between our proposed approach and existing ones. The baseline model mixture of FAs (MFA) and the MCFA are introduced as preliminaries in Section 3. In Section 4 we introduce the proposed 2L-MJFA model, while Section 5 explains how the model parameters can be estimated by the modified EM algorithm. In Section 6 we present the experimental setup and the classification results with the aid of seven datasets including a synthetic dataset and six real ones. Finally, Section 7 concludes the work. The work presented here is an extension of [17], and is based on redesigning and supplementing the experiments to support evaluations for S3 data cases, and further compare with existing methods with respect to their technical details.

2. Related work

There have been several joint learning FA based approaches [18, 19] related to our proposed method. To illustrate the distinction, we present the different alternative structures incorporated in various models in Fig.1. In particular, the model MFA [2] is the base model for what we propose. It combines DR with clustering and utilizes a subspace metric to guide cluster separation. This work is extended by MCFA [14] which assumes the factor loading of the MFA to be a common matrix that can largely reduce the involved parameters. When MCFA is used for classification, one straightforward way is to regard each class as one component, as shown in Fig.1(a). Obviously, such a setting is quite basic and not adequately flexible, since data classes may have complex distributions and modalities. Another popular variant that extends MCFA is mMCFA, shown in Fig.1(b)), where the factor loadings $\mathbf{A}_i$ are different for each class. In general, different loading matrices imply independent DR for different classes and this may be physically impractical. More importantly, mMCFA could be problematic in S3 problems, as the limited number of samples cannot support accurate learning of the loading matrices. To this end, a non-trivial model is proposed here by sharing one loading matrix for all the classes. The mPPCA method [15, 16] extends PCA to a mixture distribution model. As seen in Fig.1(d), its graphical model is quite similar to MFA with the elements of the common covariance matrix $\mathbf{D} = \sigma^2 \mathbf{I}_p$ assumed to be isotropic [20], where $\mathbf{I}_p$ is the p-



(a) MCFA (b) mMCFA

(c) 2L-MJFA (d) mPPCA

Figure 1: Comparison of different models, where $\mathbf{Y}$ denotes observed data and $\mathbf{A}$ factor loadings. (a) MCFA which is the fundamental MFA model with a common $\mathbf{A}$. (b) Mixture of MCFAs with each class consisting of a components mixture with individual local factor loadings $\mathbf{A}_i$. (c) The proposed 2L-MJFA with a global factor loading $\mathbf{A}$ shared between and within classes in the 2-layer mixture model. (d) Mixture of probabilistic PCA which is similar to MFA but with isotropic common covariance matrix.

dimensional identity matrix. For classification, each class is modeled as an mPPCA model. This method is limited due to its poor flexibility and has many redundant parameters for dealing with S3 problems.

We now analyse the parameter numbers in the different models, assuming p dimensions, q reduced dimensions from p, and m classes. Setting g mixture components in each class, the covariance matrix of each component has $N = \frac{p(p+1)}{2}$ parameters. Since mPPCA converts the diagonal covariance matrix into an isotropic one as $\mathbf{\Sigma}_i = \mathbf{W}_i\mathbf{W}_i^T + \sigma^2\mathbf{I}_p$, where factor loading $\mathbf{W}_i \in \mathbb{R}^{p\times q}$ contains $\frac{q(q-1)}{2}$ constraints, its total number of parameters is

$$N_1 = m\left(g + gp + gpq - \frac{gq(q-1)}{2}\right)$$

If either p or q is large, the number of parameters may not even be manageable with a diagonal covariance. To further reduce the parameters and accelerate

Table 1: Summary of the number of parameters for the main models.

Model:	Number of parameters:	Approximation:
mPPCA	$m[g + gp + gpq - \frac{gq(q-1)}{2}]$	$(mg + mq)p$
mMCFA	$m[pq - q^2 + p + g(1 + q + \frac{q(q+1)}{2})]$	$(m + mq)p$
2L-MJFA	$pq - q^2 + p + m[g + gq + \frac{gq(q+1)}{2}]$	$(q + 1)p$

training, the component covariance matrices of mMCFA has a factor-analytic representation $\mathbf{\Sigma}_i = \mathbf{A}\mathbf{\Omega}_i\mathbf{A}^T + \mathbf{D}$, where $\mathbf{D}$ is a diagonal matrix and $\mathbf{A}$ contains the factor loading for all the components [21]. From the orthogonality requirement, $\mathbf{A}$ has $pq - q^2$ constraints. Hence, in mMCFA the total number of parameters is reduced to

$$N_2 = m\left[pq - q^2 + p + g\left(1 + q + \frac{q(q+1)}{2}\right)\right].$$

Table 1 lists the associated parameter numbers for FA models. Since, $p >> q$ the order of the number of parameters can be approximated via the simpler form shown in the rightmost table column. It can be seen that the proposed 2L-MJFA requires the least number of parameters, which ultimately make it more suitable for dealing with S3 problems; this is also verified in the experimental results.

3. Preliminaries

As a linear model, FA decomposes a factor loading to cross a linear subspace within the covariate vector space, making factors have lower dimension than the covariates. In the following, we will first introduce MFA [22], and then we will review the fundamentals of its special case, that is MCFA [14]. Let $\mathbf{Y} \in \mathbb{R}^{n\times p}$ denote n p-dimensional vectors of feature variables generated by a linear combination of latent variables $\mathbf{Z}$. The latent variable model MFA approximates nonlinear manifolds via generating a local linear combination, relating an observation pattern to a corresponding unobservable factor vector. MFA is a directed generative model with probability π_i, with $i = 1, \ldots, g$ being the component indicator. The distribution of the difference between observations $\mathbf{y}_1, \ldots, \mathbf{y}_n$ and the g components (with means $\boldsymbol{\mu}_i$) can be defined as

$$\mathbf{y}_j - \boldsymbol{\mu}_i = \mathbf{W}_i \mathbf{Z}_{ij} + \mathbf{e}_{ij}, \quad \sum_{i=1}^{g} \pi_i = 1, \tag{1}$$
$$\mathbf{Z}_{ij} \sim \mathcal{N}(0, \mathbf{I}_q), \quad \mathbf{e}_{ij} \sim \mathcal{N}(0, \mathbf{D}_i), \quad j = 1, \ldots, n.$$

Conventionally, $\mathbf{W}_i \in \mathbb{R}^{p \times q}$ is the loading matrix which contains the factor loadings. $\mathbf{Z}_{ij}$ is a q-dimensional vector representing the unobservable factor, and $\mathbf{D}_i$ a $p \times p$ diagonal matrix with the variances of the independent noise $\mathbf{e}_{ij}$.

As a special case, the MCFA model further reduces the MFA parameters by setting up a common component factor loading $\mathbf{A} \in \mathbb{R}^{p \times q}$. Moreover, the common loading can be considered as a transformation that reduces the p-dimensional space to a latent q-dimensional one. The new model is established by rewriting Eq.(1) as

$$\mathbf{y}_j = \mathbf{A}\mathbf{Z}_{ij} + \mathbf{e}_{ij},$$
$$\mathbf{Z}_{ij} \sim \mathcal{N}(\boldsymbol{\xi}_i, \boldsymbol{\Omega}_i), \quad \mathbf{e}_{ij} \sim \mathcal{N}(0, \mathbf{D})\ . \tag{2}$$

By assuming additional constraints, we can obtain

$$\boldsymbol{\mu}_i = \mathbf{A}\boldsymbol{\xi}_i, \quad \boldsymbol{\sigma}_i^2 = \mathbf{A}\boldsymbol{\Omega}_i\mathbf{A}^T + \mathbf{D}, \quad \mathbf{D}_i = \mathbf{D}, \quad \mathbf{W}_i = \mathbf{A}\mathbf{K}_i. \tag{3}$$

In the above, $\boldsymbol{\xi}_i$ is a q-dimensional vector and $\boldsymbol{\Omega}_i$ is a $q \times q$ positive definite matrix. Differently from MFA, the independent noise variance matrix $\mathbf{D}$ is a global parameter instead of the local one $\mathbf{D}_i$. For an observed random sample $\mathbf{y}_1, \ldots, \mathbf{y}_n$ the MCFA model becomes a mixture of Gaussians with constrained mean and covariance as defined in Eq.(3), and is given by

$$\begin{aligned} P(\mathbf{y}_j; \boldsymbol{\theta}_i) &= \sum_{i=1}^{g} \pi_i \prod_{j=1}^{n} \mathcal{N}(\mathbf{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2) \\ &= \sum_{i=1}^{g} \pi_i \prod_{j=1}^{n} \mathcal{N}(\mathbf{y}_j; \mathbf{A}\boldsymbol{\xi}_i, \mathbf{A}\boldsymbol{\Omega}_i\mathbf{A}^T + \mathbf{D}), \end{aligned} \tag{4}$$

where $\boldsymbol{\theta}_i = \{\pi_i, \mathbf{A}, \boldsymbol{\xi}_i, \boldsymbol{\Omega}_i, \mathbf{D}\}_i^g$ are the model parameters. Each component can be modeled through a Gaussian distribution $\mathcal{N}(\mathbf{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i^2)$. Given the mixture of g components, with $\boldsymbol{\omega}_{ij}$ denoting the binary component indicator that are one if and only if the j^{th} object belongs to the i^{th} component, the posterior can be expressed with Bayes theorem as

$$P(\omega_i \mid \mathbf{y}_j; \boldsymbol{\theta}) = \tau_i(\mathbf{y}_j; \boldsymbol{\theta}_i) = \frac{\pi_i \mathcal{N}(\mathbf{y}_j; \boldsymbol{\theta}_i)}{\sum_{h=1}^{g} \pi_h \mathcal{N}(\mathbf{y}_j; \boldsymbol{\theta}_h)}. \tag{5}$$

Since the latent variables $\mathbf{Z}_{i1}, \ldots, \mathbf{Z}_{in}$, are distributed independently as in Eq.(2), the probability density function is $P(\mathbf{Z}_{ij} \mid \boldsymbol{\omega}_{ij}) = \mathcal{N}(\mathbf{Z}_{ij} \mid \boldsymbol{\xi}_i, \boldsymbol{\Omega}_i)$.

For the training stage, the model parameters can be determined via maximum-likelihood using the EM algorithm [23, 24]. The likelihood an log-likelihood of the model are given by

$$\begin{aligned}
\mathcal{L}(\mathbf{y}) = & \prod_{j=1}^{n} \sum_{i=1}^{g} P(\mathbf{y}_j \mid \mathbf{Z}_{ij}, \boldsymbol{\omega}_{ij}) P(\mathbf{Z}_{ij} \mid \boldsymbol{\omega}_{ij}) P(\boldsymbol{\omega}_{ij}), \\
\log \mathcal{L}(\theta) = & \sum_{i=1}^{g} \sum_{j=1}^{n} \boldsymbol{\omega}_{ij} \Big\{ \log \pi_i + \log \mathcal{N}(\mathbf{y}_j; \mathbf{A}\mathbf{u}_{ij}, \mathbf{D}) \\
& \qquad + \log \mathcal{N}(\mathbf{Z}_{ij}; \boldsymbol{\xi}_i, \boldsymbol{\Omega}_i) \Big\}.
\end{aligned} \tag{6}$$

Therefore, the parameters $\boldsymbol{\theta}$ can be optimized by maximizing the expected log-likelihood $\mathbb{E}_{\tau_i}[\log \mathcal{L}(\boldsymbol{\theta})]$. The detailed algorithm can be found in [25].

4. Two-layer mixture of factor analyzers with joint factor loading

Let us consider the construction of a 2L-MJFA with two hidden layer factors, with these factors sharing a common factor loading. For classification, the observation data are known as $\mathbf{Y} = [Y_1; \ldots; Y_m]$, where $\mathbf{Y}_l = [\mathbf{y}_1^l; \ldots; \mathbf{y}_{l_n}^l]$, and $l = 1, \ldots, m$ indicates all the data of the l^{th} class. In our model, the 1^{st} layer defines a normal mixture of factor analyzers with common loading, where each component represents a class, as

$$\mathbf{y}_j^l = \mathbf{A}\mathbf{U}_j^l + \mathbf{e}_j^l, \qquad j = 1, \ldots, l_n, \quad \sum_{l=1}^{m} l_n = n.$$

In the above, l_n denotes the n^{th} observation belonging to l^{th} class, and $\mathbf{U}_j^l$ denotes the hidden variables. $\mathbf{A} \in \mathbb{R}^{p \times q}$ is the joint factor loading to fit all classes of observations, which can also be considered to be the transformation matrix that projects each pattern to a q-dimensional latent space. $\mathbf{e}_j^l$ denotes the Gaussian noise term for the l^{th} class.

The 2^{nd} layer of 2L-MJFA representing each class, consists of an unspecified number of mixtures. The key point here is that the joint factor loading $\mathbf{A}$ is also used as a common loading that is shared across all the components in each class. Then all the observations can be generated by a joint learning model with latent variables $\mathbf{Z}_{ij} \sim \mathcal{N}(\boldsymbol{\xi}_i, \boldsymbol{\sigma}_i^2)$ of all classes.

For the observation vectors $\mathbf{y}_j^l$ belonging to each class l, the model can then be described as

$$\mathbf{y}_j^l = \mathbf{A} \sum_{i=1}^{g} \mathbf{Z}_{ij}^l + \mathbf{e}_j^l,$$

where $j = 1, \ldots, l_n$, and $i = 1, \ldots, g$. l_n denotes the n^{th} observation belonging to l^{th} class, and $\mathbf{e}_j^l$ the random noise distributed independently under $\mathcal{N}(0, \mathbf{D})$, where $\mathbf{D}$ is diagonal. This novel setting implies that each specific class is assumed to be an MCFA model, whereas a joint factor loading exists for all the MCFAs across all data classes. Specifically, the model shares a joint factor loading for all the classes and this is potentially beneficial to both feature extraction and classification, especially in S3 situations.

We now calculate the total number of parameters involved in 2L-MJFA. Since we share a single loading matrix across all the components, the total number of parameters is

$$N_3 = pq - q^2 + p + mg \left[1 + q + \frac{q(q+1)}{2} \right],$$

where $pq - q^2$ is the number of parameters in $\mathbf{A}$, and p the parameters of the diagonal matrix $\mathbf{D}$. The mMCFA offers a great reduction in the parameters of the loading $\mathbf{A}$ for each component. Compared with mMCFA, the proposed model significantly reduces the parameter number by $(m-1)(pq - q^2 + p)$.

5. Optimization via a modified EM algorithm

The proposed 2L-MJFA model is composed of two layers of mixture of Gaussians. The overall distribution for the mixture of mixtures is the joint distribution of their components given as

$$P(\mathbf{y}_j^l;\boldsymbol{\theta}) = \sum_{l=1}^{m} \pi_l \prod_{j=1}^{l_n} P(\mathbf{y}_j^l;\boldsymbol{\theta}), \tag{7}$$

where $\boldsymbol{\theta} = \{\pi_i, \mathbf{A}, \boldsymbol{\xi}_i^l, \boldsymbol{\Omega}_i^l, \mathbf{D}\}$. Actually, the 2^{nd} layer of each class is an MCFA model, which can be easily written as the multivariate Gaussian distribution of Eq.(4). For inference, the conditional expectation of the component indicators $\boldsymbol{\omega}_{ij}^l$ with $i = 1, \ldots, g$ and $l = 1, \ldots, m$, can be regarded as the posterior probability $P_{\boldsymbol{\theta}}\{\boldsymbol{\omega}_{ij}^l = 1 \mid \mathbf{y}_j^l\}$, implying that $\mathbf{y}_j^l$ belongs to the i^{th} component of class l. With the above definitions, we obtain the conditional distribution $P(\mathbf{y}_j^l \mid \mathbf{U}_{ij}^l) = \mathcal{N}(\mathbf{y}_j^l \mid \mathbf{A}\mathbf{U}_{ij}^l, \boldsymbol{\theta})$. The posterior over all components can then be obtained as

$$\mathbb{E}_\theta\{\boldsymbol{\omega}_i^l \mid \mathbf{y}_j^l\} = Pr_{\boldsymbol{\theta}}\{\boldsymbol{\omega}_{ij}^l = 1 \mid \mathbf{y}_j^l\} = \tau_i^l(\mathbf{y}_j^l;\boldsymbol{\theta}), \tag{8}$$

where

$$\tau_i^l(\mathbf{y}_j^l;\boldsymbol{\theta}) = \frac{\pi_l P(\mathbf{y}_j^l;\boldsymbol{\theta})}{\sum_{h=1}^m \pi_h P(\mathbf{y}_j^l;\boldsymbol{\theta})}.$$

Maximum likelihood learning of 2L-MJFA can be conducted with a modified EM algorithm. Within the modified EM framework, the global log-likelihood function of the model is given by

$$\begin{aligned} \log L_l(\boldsymbol{\theta}) = \sum_{l=1}^{m}\sum_{i=1}^{g}\sum_{j=1}^{n} \quad & \boldsymbol{\omega}_{ij}^l \Big\{ \log \pi_l + \log \phi(\mathbf{y}_j^l; \mathbf{A}\mathbf{U}_{ij}^l, \mathbf{D}) \\ & + \log \phi(\mathbf{U}_{ij}^l; \boldsymbol{\xi}_{ij}^l, \boldsymbol{\Omega}_{ij}^l) \Big\}, \end{aligned} \tag{9}$$

where

$$\phi(\mathbf{y}_j^l;\boldsymbol{\theta}) = \sum_{i=1}^{g} \pi_i^l \mathcal{N}(\mathbf{y}_j^l;\boldsymbol{\theta}).$$

Differently from the alternating expectation – conditional maximization algorithm (AECM) [21], the M-step of the modified EM algorithm is turned into two layer loops. The outer loop is used to update the global parameters $\mathbf{A}$ and $\mathbf{D}$, and the other parameters within each specific class are updated in the inner loop. The training of the above two layers alternate, so that a local optimum could be finally achieved. The overall EM training procedure is summarized in Algorithm 1, and specifics for each stage are explained in the following subsections.

Algorithm 1: EM learning for 2L-MJFA.

Input : Training data $\mathbf{Y} = [Y_1; \ldots; Y_m], \mathrm{Y} \in R^{n \times p}$.

Output : Optimal values of parameters $\boldsymbol{\theta}$.

Initialization: Set $\theta = \{\pi, \mathbf{A}, \boldsymbol{\xi}, \boldsymbol{\Omega}, \mathbf{D}\}$, and evaluate the initial value of the log-likelihood.

Repeat

- **E-step :**
  - Exploit the current parameter values to approximate the posterior expectations with Eqs.(10,11): $\mathbb{E}(\mathbf{Z} \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l)$ and $\mathbb{E}(\mathbf{Z}\mathbf{Z}^T \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l)$.
- **for** $l = 1$ *to* m **do**
  - **M-step :**
    - Update $\mathbf{A}$ and $\mathbf{D}$.
    - Re-estimate the parameters $\mathbf{A}, \mathbf{D}$ using the current responsibilities with Eqs.(13,14), by solving a set of liner equations: $\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)})}{\partial \mathbf{A}} = 0$, $\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)})}{\partial \mathbf{D}} = 0$.
    - Update $\{\pi, \boldsymbol{\xi}, \boldsymbol{\Omega}\}$.
    - **for** $i = 1$ *to* g **do**
      - Re-estimate the parameters $\pi_i^l, \boldsymbol{\xi}_i^l, \boldsymbol{\Omega}_i^l$ by solving the equations $\pi_i^{(k+1)} = \frac{1}{n_l} \sum_{j=1}^{l_n} \tau_{ij}^{(k)}$, $\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\xi}_i} = 0$ and $\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\Omega}_i} = 0$ for each class.

Until *Convergence*

5.1. E-step

In this step, Eq.(5) is used to compute the posterior over the latent variables. Given the current setting of the model parameters, the expectations of the hidden variables $\mathbb{E}(\mathbf{Z} \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l)$ and $\mathbb{E}(\mathbf{Z}\mathbf{Z}^T \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l)$ are easily verified as the following derivations for all the data points $j = 1, \ldots, l_n$ and mixture components $i = 1, \ldots, g$ can be produced as

$$\mathbb{E}(\mathbf{Z} \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l) = \boldsymbol{\xi}_i^l + \boldsymbol{\gamma}_i^T \mathbf{y}_{ij}, \tag{10}$$

and

$$\mathbb{E}(\mathbf{Z}\mathbf{Z}^T \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l) = (\mathbf{I}_q - \boldsymbol{\gamma}_i^T \mathbf{A})\boldsymbol{\Omega}_{ij}^l + \mathbb{E}(\mathbf{Z} \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l)\mathbb{E}(\mathbf{Z} \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l)^T, \tag{11}$$

where

$$\begin{aligned} \mathbf{y}_{ij} &= \mathbf{y}_j^l - \mathbf{A}\boldsymbol{\xi}_i^l, \\ \boldsymbol{\gamma}_i &= (\mathbf{A}\boldsymbol{\Omega}_i^l \mathbf{A}^T + \mathbf{D})^{-1} \mathbf{A}\boldsymbol{\Omega}_i. \end{aligned}$$

For the iteration of each class, $\mathbf{Q}(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)})$ denotes the conditional expectation of Eq.(7) as

$$\mathbf{Q}(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)}) = P(\mathbf{Z}^{(k)} \mid \mathbf{y}^{(k)}; \boldsymbol{\theta}), \tag{12}$$

given the observed data $\mathbf{y}$ and $\boldsymbol{\theta}^{(k)}$. Denoting the posterior $\tau_{ij}^{(k)} = \tau_i^l(\mathbf{y}_j^l; \boldsymbol{\theta}^{(k)})$, we can transform Eq.(12) as

$$\begin{aligned} \mathbf{Q}(\boldsymbol{\theta}; \boldsymbol{\theta}^{(k)}) = \sum_{i=1}^{g} \sum_{j=1}^{l_n} \tau_{ij}^{(k)} \Big\{ &[\log \pi_i^l + \mathbb{E}_{\boldsymbol{\theta}^{(k)}} [\log \mathcal{N}(\mathbf{y}_j^l; \mathbf{A}\mathbf{Z}_{ij}^l, \mathbf{D}) | \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^l = 1] \\ &+ \mathbb{E}_{\boldsymbol{\theta}^{(k)}} [\log \mathcal{N}(\mathbf{Z}_{ij}^l; \boldsymbol{\xi}_i^l, \boldsymbol{\Omega}_i^l) | \mathbf{y}_j, \boldsymbol{\omega}_{ij}^l = 1] \Big\}. \end{aligned}$$

5.2. M-step

In subsequent step, the updated estimates of the global parameters can be obtained by taking the partial derivatives of expectation log-likelihood function for each parameter. The joint factor loading is updated as

$$\mathbf{A}^{(k+1)} = \left(\sum_{l=1}^{m} \sum_{i=1}^{g} \mathbf{A}_{li(1)}^{(k)} \right) \left(\sum_{l=1}^{m} \sum_{i=1}^{g} \mathbf{A}_{li(2)}^{(k)} \right)^{-1}, \tag{13}$$

Algorithm 2: Classification procedure for 2L-MJFA.

Input: A training set with m classes $[\mathbf{Y}_1; \dots; \mathbf{Y}_m]$ and a test set $\mathbf{T} \in \mathbb{R}^{N \times P}$.

Training phase :

Initialize the global parameters $\mathbf{A}, \mathbf{D}$ based on all the training data.

Divide each $\mathbf{Y}_l$, for $l = 1, \dots, m$ into g components randomly and then initialize the local parameters $\pi_i, \boldsymbol{\xi}_i, \boldsymbol{\Omega}_i$.

Repeat

for $l = 1$ *to* m **do**

Estimate the probability of data generated by each component in Eq.(7) and the posterior probability $P_{\boldsymbol{\theta}}\{\boldsymbol{\omega}_{ij}^l = 1 | \mathbf{T}_j\}$, for $j = 1, \dots, N$ that $\mathbf{T}_j$ belongs to the i^{th} component by each class in Eq.(8).

for $i = 1$ *to* g **do**

Use the alternate EM algorithm, and update local parameters by calculating the expectation of log-likelihood in Eq.(12) of each class.

Compute the log-likelihood value $L_l(\boldsymbol{\theta})$ using Eq.(9).

Until $\mathrm{L}_l(\boldsymbol{\theta})^{(new)} - \mathrm{L}_l(\boldsymbol{\theta}) <$ *threshold value*

Testing phase :

Compute the posterior probabilities $\tau_l(\mathbf{T}_j; \boldsymbol{\theta})$ of each class with test data.

Assign each test data point $\mathbf{T}_j$ to the l class for which $\tau_l(\mathbf{T}_j; \boldsymbol{\theta}) \geq \tau_h(\mathbf{T}_j; \boldsymbol{\theta})$ for $h = 1, \dots, m$ with $h \neq l$.

where

$$\mathbf{A}_{li(1)}^{(k)} = \sum_{j=1}^{l_n} \tau_{ij}^{(k)} \left\{ \mathbf{y}_j^l \mathbb{E}^{(k)}(\mathbf{Z} \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^{l(k)}) \right\},$$

$$\mathbf{A}_{li(2)}^{(k)} = \sum_{j=1}^{l_n} \tau_{ij}^{(k)} \left\{ \mathbb{E}^{(k)}(\mathbf{Z}\mathbf{Z}' \mid \mathbf{y}_j^l, \boldsymbol{\omega}_{ij}^{l(k)}) \right\}.$$

The updated estimates of the common diagonal covariance matrix can then be written as

$$\mathbf{D}^{(k+1)} = \frac{1}{n} diag\Big[\sum_{l=1}^{m} \sum_{j=1}^{l_n} \tau_{ij}^{(k)} (\mathbf{D}_1^{(k)} + \mathbf{D}_2^{(k)})\Big], \tag{14}$$

where

$$\mathbf{D}_1^{(k)} = \mathbf{D}^{(k)}(\mathbf{I}_p - \boldsymbol{\beta}^{(k)}),$$
$$\mathbf{D}_2^{(k)} = \boldsymbol{\beta}^{(k)^T}(\mathbf{y}_{ij}^{(k)})(\mathbf{y}_{ij}^{(k)})^T \boldsymbol{\beta}^{(k)},$$
$$\boldsymbol{\beta}^{(k)} = \left(\mathbf{A}^{(k)}\boldsymbol{\Omega}^{(k)}\mathbf{A}^{(k)^T} + \mathbf{D}^{(k)}\right)^{-1} \mathbf{D}^{(k)}.$$

For each class l, the updated estimates $\pi_i^{(k+1)}$, $\boldsymbol{\xi}_i^{(k+1)}$ and $\boldsymbol{\Omega}_i^{(k+1)}$ can be obtained by calculating the equations $\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\xi}_i} = 0$, $\frac{\partial Q(\boldsymbol{\theta};\boldsymbol{\theta}^{(k)})}{\partial \boldsymbol{\Omega}_i} = 0$. Specifically, it is easy to verify that $\pi_i^{(k+1)} = \frac{1}{n_l} \sum_{j=1}^{l_n} \tau_{ij}^{(k)}$, for $i = 1, \dots g$, where n_l denotes the number of observations in l_{th}class. The local parameter updates can be obtained via the following

$$\boldsymbol{\xi}_i^{(k+1)} = \boldsymbol{\xi}_i^{(k)} + \frac{\sum_{j=1}^{l_n} \tau_{ij}^{(k)} \boldsymbol{\varphi}^{(k)}}{\sum_{j=1}^{l_n} \tau_{ij}^{(k)}},$$
$$\boldsymbol{\Omega}_i^{(k+1)} = \frac{\sum_{j=1}^{l_n} \tau_{ij}^{(k)} \boldsymbol{\varphi}^{(k)} \boldsymbol{\varphi}^{(k)^T}}{\sum_{j=1}^{l_n} \tau_{ij}^{(k)}} + (\mathbf{I}_q - \boldsymbol{\varphi}^{(k)})\boldsymbol{\Omega}_i^{(k)},$$
$$\boldsymbol{\varphi}^{(k)} = \boldsymbol{\gamma}_i^{(k)^T} \mathbf{y}_{ij}^{(k)}.$$

Algorithm 2 summarizes the overall classification procedure.

6. Experiments and Results

To demonstrate the effectiveness of our proposed algorithm, we conduct extensive experiments on a variety of datasets. We compare our two-layer mixture approach with three other competitive methods. Specifically, we compare it with mMCFA, mixture of PPCA (mPPCA), and the independent learning approaches of PCA followed by GMM (PCA-GMM), and LDA followed by GMM (LDA-GMM)[1]. Unlike hard assignment methods (e.g. k-means), GMM is a soft assignment method which gives the probability that the data points are assigned to each class, rather than just giving a definitive class membership [26]. Obtaining a probability is beneficial as it provides confidence for the results. The used datasets include a synthetic one, an ordinary one, and five S3 datasets. We report the error rate (ERR) of the classification in terms of different reduced dimensionalities for the various algorithms on the test data. All the experimented methods are implemented in the MATLAB platform.

6.1. Synthetic dataset

To illustrate the advantage of the joint learning in the proposed model, we generate a synthetic data to visualize the obtained subspaces for PCA, MCFA and the 2L-MJFA. The synthetic dataset consists of 82 classes of $32-$dimensional samples. For each class, the first two dimensions are randomly generated by a multivariate normal distribution with means and covariance set to

$$\mu_1 = (3.2875, 3.4905)^T, \qquad \mu_2 = (2.9185, 2.9732)^T,$$
$$\Sigma_1 = \begin{pmatrix} 23.2368 & 19.2956 \\ 19.2956 & 19.8985 \end{pmatrix}, \quad \Sigma_2 = \begin{pmatrix} 5.0030 & 0.8919 \\ 0.8919 & 4.4236 \end{pmatrix}.$$

The other 30-dimensions are generated as random Gaussian noise.

The obtained 2-dimensional subspaces are visualized in Fig.2. The top-left of the figure shows the ground truth samples without the additional 30-dimensional noisy features. It can be clearly seen that the class denoted by label 1 consists of two modalities. The proposed 2L-MJFA shows to perform better than the other two, as its subspace demonstrates a much better separability than PCA and MCFA. The mPPCA does not map all the data in a subspace, since the

[1]PCA or LDA are firstly used to perform DR and then a GMM is used for the classification.

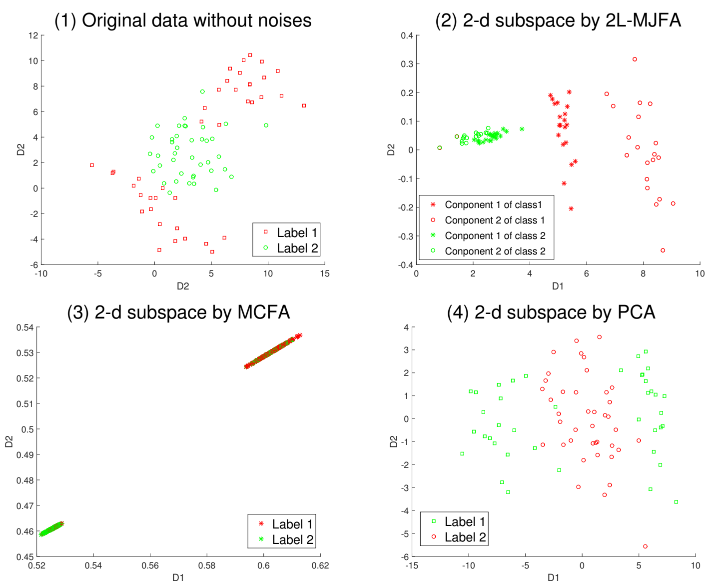


Figure 2: Visualization of DR for 2L-MJFA, MCFA, and PCA on simulated data, where (1) is the ground truth. Different patterns represent different classes, and different shapes within the same grey scale indicate different class modalities.

approach is used to classification by building an mPPCA model of each class, which means that the patterns for different classes are mapped into different subspaces. Also, LDA can generate subspaces up to $m-1$ dimensions, which is one dimension for the current dataset.

6.2. User knowledge data

The employed User Knowledge dataset describes students' knowledge status about the subject of Electrical DC Machines [27]. This dataset consists of 403 training samples and 206 test samples. Each sample is of 40 dimensions with 5 being attribute information, plus 35 random noisy features. The class labels correspond to four student knowledge levels. We compare the 2L-MJFA and other mixture joint learning methods against different reduced dimensionalities ranging from 1 to 20.

We report the comparative results in Table 2. We can see that the mixture joint learning methods 2L-MJFA and mMCFA provide the lowest error rates.

Table 2: Error rate comparison for various dimensions, for the User Knowledge dataset.

Dimension:	1	3	5	10	15	20
2L-MJFA	0.1214	0.0689	0.0414	0.0620	0.0620	0.0620
mMCFA	0.2276	0.0552	0.0896	0.0758	0.1517	0.2827
mPPCA	0.3172	0.2897	0.2690	0.1931	0.1586	0.0897
PCA-GMM	0.6621	0.2483	0.2345	0.1214	0.1931	0.2966
LDA-GMM	0.4000	0.3724	-	-	-	-

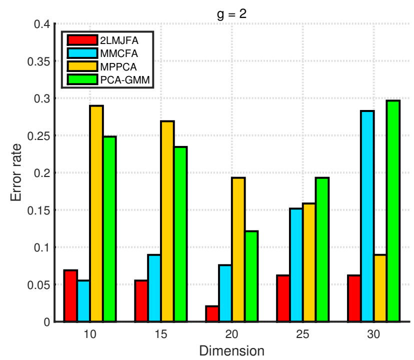


Figure 3: Error rate comparison for the User Knowledge dataset.

In particular, when the dimensionality is reduced to 5 (the actual dimension), 2L-MJFA yields the best performance with the error rate being 0.0414. This is significantly lower than mMCFA, mPPCA and PCA-GMM. LDA-GMM just allows to reduce dimensionality to $1-3$, since this dataset has $m=4$ classes. From the results, LDA does not provide an optimized subspace for test data. To better illustrate the performance, we also plot the results in Fig.3, where it can be seen that 2L-MJFA outperforms the other algorithms in most cases.

6.3. Small sample size datasets

In this subsection, we compare the proposed 2L-MJFA with the various other algorithms across five S3 datasets.

Experimental Setup. We evaluate the performance of the various algorithms by using a 5-fold cross validation on the five S3 datasets, which are WDBC, WPBC, ULC, LSVT and BT. To make the problems more challenging, we

Table 3: Summary of S3 datasets.

Dataset:	Training samples:	Test samples:	Dimensions:	Classes:
WDBC	114	455	60	2
WPBC	38	156	33	2
ULC	77	273	148	3
LSVT	56	42	309	2
BT	81	24	39	6

intentionally use one of the five partitions as the training set, while the remaining four partitions as the testing set. The average error rate on the test sets is then reported for varying mixture numbers and reduced dimensionalities. Table 3 summarizes the statistics of these five S3 datasets. As seen in the table, the number of dimensions are sometimes larger than the number of training samples (e.g., in ULC and LSVT).

6.3.1. Breast cancer Wisconsin dataset

This dataset contains two subsets, the Wisconsin diagnostic breast cancer (WDBC) and the Wisconsin prognostic breast cancer (WPBC) [28, 29]. WDBC contains 569 instances which are divided into the two diagnostic predictions of benign and malignant. The 60 attributes consist of 30 real-valued input features and 30 additional Gaussian noise features. WPBC contains 194 instances, which record two classes of patients, that is being recurrent or not post-surgical.

Wisconsin diagnostic breast cancer (WDBC). Table 4 shows the error rate comparison from reducing the dimensions from 10 to 30 and setting each class to $g = 2$ to 5 mixture components for different subspaces (DIM). For LDA-GMM, the dimensionality is just allowed to reduce to 1, because there are 2 classes in these two datasets. We can find that the error rate of 2L-MJFA decreases as the number of mixture components increases. For clarity, we also plot the results in Fig.4, where it can be observed that 2L-MJFA achieves the significantly lowest error rate 0.0404 when the dimension is reduced to 30 and the number of components is set to 5. The best result of the competitors is just 0.0279 given by LDA-GMM.

Table 4: Error rate comparison for the WDBC dataset.

WDBC						
DIM	g	2L-MJFA	mMCFA	mPPCA	PCA-GMM	LDA-GMM
1	2	0.1023±0.02	0.0703±0.02	0.2846±0.03	0.0935 ±0.03	0.0350±0.01
	3	0.1010±0.01	0.0686±0.02	0.2509±0.03	0.0935±0.03	0.0282±0.01
	4	0.1022±0.01	0.0703±0.03	0.2778±0.04	0.0935±0.03	0.0334±0.01
	5	0.1076±0.01	0.0705±0.02	0.2759±0.01	0.0935±0.03	0.0334±0.01
10	2	0.0746±0.02	0.0742±0.01	0.3202±0.01	0.1502±0.12	-
	3	0.0707±0.02	0.0861±0.02	0.3019±0.02	0.1528±0.15	-
	4	0.0716±0.02	0.0817±0.02	0.2465±0.05	0.1571±0.12	-
	5	0.0441±0.03	0.0842±0.02	0.2065±0.03	0.1600±0.11	-
15	2	0.0698±0.02	0.0707±0.01	0.3212±0.03	0.2182±0.15	-
	3	0.0689±0.02	0.0830±0.01	0.3041±0.03	0.2050±0.10	-
	4	0.0716±0.03	0.0922±0.02	0.3295±0.04	0.2114±0.11	-
	5	0.0737±0.03	0.0963±0.03	0.2917±0.03	0.2147±0.09	-
20	2	0.0755±0.02	0.0703±0.02	0.3448±0.05	0.2406±0.12	-
	3	0.0755±0.03	0.0707±0.01	0.3348±0.06	0.2343±0.08	-
	4	0.0645±0.02	0.0914±0.04	0.3005±0.06	0.2536±0.08	-
	5	0.0641±0.01	0.0833±0.03	0.2956±0.02	0.2749±0.09	-
25	2	0.0680±0.02	0.0707±0.01	0.3405±0.02	0.2481±0.11	-
	3	0.0597±0.03	0.0712±0.02	0.3199±0.07	0.2775±0.06	-
	4	0.0505±0.01	0.0776±0.02	0.3097±0.04	0.3189±0.07	-
	5	0.0479±0.04	0.0782±0.02	0.2917±0.02	0.3633±0.02	-
30	2	0.0417±0.02	0.0707±0.01	0.3110±0.03	0.2938±0.07	-
	3	0.0483±0.01	0.0743±0.02	0.2935±0.02	0.3229±0.09	-
	4	0.0422±0.01	0.0738±0.02	0.3053±0.02	0.3628±0.02	-
	5	0.0404±0.04	0.0681±0.01	0.2987±0.02	0.3606±0.03	-

Wisconsin prognostic breast cancer (WPBC). The results for this comparison are shown in Table 5 and Fig.5. We can clearly observe that the 2L-MJFA again achieves the overall best performance. In particular, the 2L-MJFA achieves the lowest error rate 0.1493 when the dimension is reduced to 25; this is significantly lower than the error of 0.1702 from MCFA.

6.3.2. Urban land cover dataset (ULC)

The ULC dataset contains nine types of urban land cover from high resolution aerial imagery [30, 31]. In this experiment, for simplicity, we only extract three types of experimental data, that is building, concrete, and grass. The

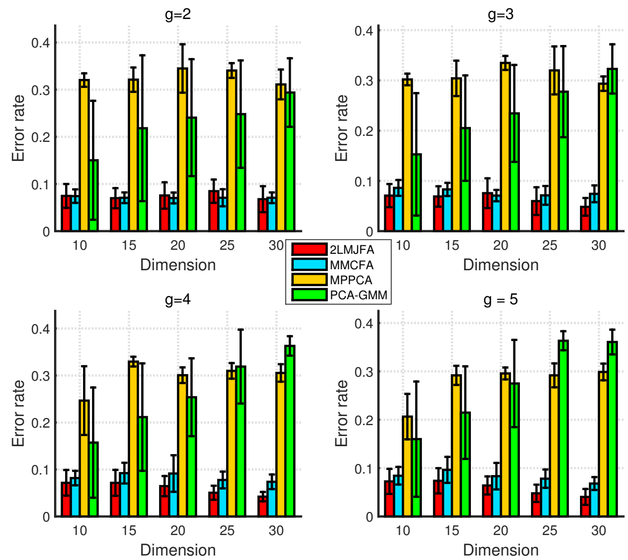


Figure 4: Error rate comparison for the WDBC dataset.

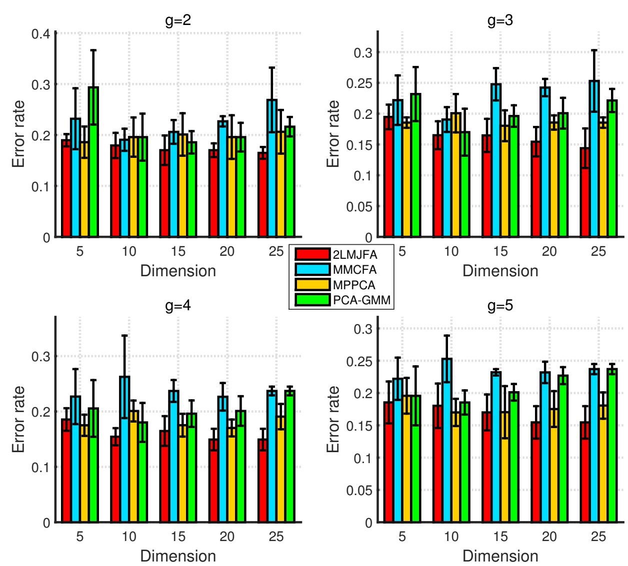


Figure 5: Error rate comparison for the WPBC dataset.

Table 5: Error rate comparison for the WPBC dataset.

WPBC						
DIM	g	2L-MJFA	mMCFA	mPPCA	PCA-GMM	LDA-GMM
1	2	0.2498±0.03	0.2943±0.10	0.3351±0.07	0.25644±0.03	0.2479±0.05
	3	0.2621±0.03	0.3045±0.14	0.2869±0.11	0.25644±0.03	0.2166±0.02
	4	0.2459±0.02	0.3947±0.12	0.2631±0.03	0.25644±0.03	0.2166±0.04
	5	0.2604±0.03	0.2887±0.13	0.2730±0.03	0.25644±0.03	0.1860±0.03
5	2	0.1896±0.01	0.2319±0.06	0.1859±0.03	0.2935±0.07	-
	3	0.1946±0.02	0.2219±0.04	0.1855±0.01	0.2318±0.04	-
	4	0.1854±0.02	0.2269±0.04	0.1751±0.02	0.2055±0.05	-
	5	0.1854±0.03	0.2220±0.03	0.2250±0.03	0.1956±0.05	-
10	2	0.1793±0.02	0.1906±0.02	0.1929±0.04	0.1957±0.05	-
	3	0.1649±0.02	0.1904±0.02	0.2007±0.03	0.1700±0.02	-
	4	0.1544±0.01	0.2625±0.07	0.2009±0.02	0.1802±0.04	-
	5	0.1802±0.02	0.2528±0.04	0.2000±0.02	0.1853±0.02	-
15	2	0.1700±0.03	0.2060±0.02	0.2010±0.04	0.1856±0.02	-
	3	0.1647±0.02	0.2477±0.02	0.1804±0.03	0.1961±0.02	-
	4	0.1647±0.02	0.2370±0.02	0.1752±0.02	0.1960±0.02	-
	5	0.1699±0.02	0.2320±0.00	0.1750±0.04	0.2011±0.01	-
20	2	0.1700±0.01	0.2268±0.01	0.1959±0.04	0.1957±0.03	-
	3	0.1544±0.02	0.2423±0.01	0.1856±0.01	0.2007±0.02	-
	4	0.1493±0.02	0.2265±0.02	0.1702±0.02	0.2009±0.03	-
	5	0.1545±0.02	0.2319±0.02	0.2000±0.03	0.2267±0.01	-
25	2	0.1648±0.01	0.2687±0.06	0.2063±0.01	0.2163±0.02	-
	3	0.1493±0.02	0.2531±0.04	0.1855±0.01	0.2214±0.06	-
	4	0.1493±0.02	0.2370±0.01	0.1907±0.02	0.2370±0.01	-
	5	0.1545±0.02	0.2370±0.01	0.1806±0.02	0.2370±0.01	-

number of components g are assumed to be between 2 and 5.

Table 6 reports the results across different dimensionalities ranging from 10 to 30 (1 to 2 for LDA-GMM). The best result of 0.1392 is achieved by 2L-MJFA model, for 30 dimensions and 5 components. The other methods perform worse, especially as the numbers of components and dimensions increase. mMCFA achieves better than the remaining methods. The errors are also summarized in Fig.6.

Table 6: Error rate comparison for the ULC dataset.

ULC						
DIM	g	2L-MJFA	mMCFA	mPPCA	PCA-GMM	LDA-GMM
2	2	0.5108	0.1209	0.5128	0.3301	0.6557
10	2	0.1319	0.1355	0.4945	0.2491	-
	3	0.1282	0.1282	0.1832	0.2015	-
	4	0.1355	0.1502	0.3736	0.2564	-
	5	0.1245	0.1319	0.2418	0.2418	-
15	2	0.1172	0.3077	0.3846	0.3700	-
	3	0.1172	0.2234	0.3773	0.3846	-
	4	0.1392	0.2381	0.2418	0.3773	-
	5	0.1099	0.1722	0.3223	0.4139	-
20	2	0.1209	0.4725	0.3773	0.3883	-
	3	0.1209	0.3919	0.3443	0.4139	-
	4	0.1245	0.3883	0.3883	0.3956	-
	5	0.1172	0.3004	0.3516	0.4396	-
25	2	0.1172	0.4579	0.3114	0.4066	-
	3	0.1392	0.3150	0.2454	0.4176	-
	4	0.1319	0.3810	0.3480	0.4066	-
	5	0.1319	0.4029	0.2930	0.4432	-
30	2	0.1245	0.4286	0.4249	0.4432	-
	3	0.1209	0.2454	0.3443	0.4396	-
	4	0.1429	0.3883	0.2527	0.4505	-
	5	0.1392	0.3883	0.3077	0.4945	-

6.3.3. LSVT voice rehabilitation dataset (LSVT)

The LSVT contains 98 instances with 309 attributes and is used for evaluating whether a phonation considered acceptable or not after voice rehabilitation [32]. The results of Table 7 are reported for different dimensions between 5 and 20 (1 for LDA-GMM). It can be seen, that mMCFA and mPPCA achieve their best performance when the dimensionality is reduced to 10. When the dimensions increase, the performance of different algorithms deteriorates quickly due to a more pronounced S3 problem. The proposed 2L-MJFA model again achieves the lowest error rate of 0.1792 (when the dimension is set to 20). Fig.7 summarizes these errors.

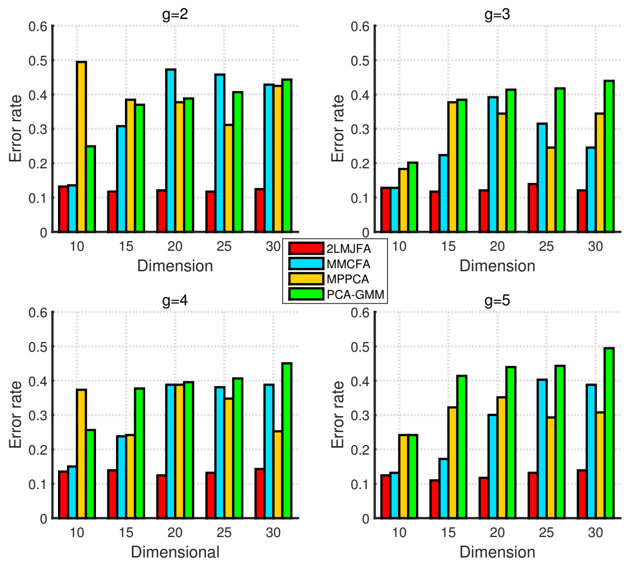


Figure 6: Error rate comparison for the ULC dataset.

Table 7: Error rate comparison for the LSVT dataset.

LSVT					
DIM	2L-MJFA	mMCFA	mPPCA	PCA-GMM	LDA-GMM
1	0.3171±0.07	0.2897±0.12	0.3731±0.07	0.4019pm0.06	0.4246±0.04
5	0.2143±0.08	0.2103±0.10	0.2143±0.04	0.2659±0.07	-
10	0.2023±0.06	0.1980±0.07	0.1964±0.06	0.2698±0.03	-
15	0.1984±0.06	0.2421±0.06	0.2183±0.06	0.2857±0.05	-
20	0.1792±0.06	0.2659±0.05	0.2857±0.03	0.2857±0.06	-

6.3.4. Breast tissue dataset (BT)

This dataset [33] contains 106 objects described by 9 features. For each object, a group of features are selected from excised breast tissue samples using electrical impedance measurement. Six major diagnostic classes are involved that consist of 4 normal breast tissues: connective, glandular, Fibro-adenoma and adipose tissue, as well as 2 pathological tissues, that is: mastopathy and carcinoma. We augment the features to 39 dimensions with random Gaussian noise, in order to accentuate the S3 effect. We report the results across different dimensionalities ranging from 2 to 9 (2 to 5 for LDA-GMM) and different

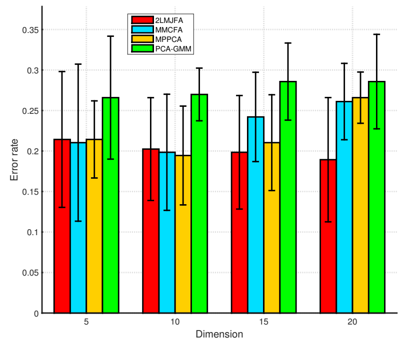


Figure 7: Error rate comparison for the LSVT dataset.

Table 8: Error rate comparison for the BT dataset.

		BT				
g	DIM	2L-MJFA	mMCFA	mPPCA	PCA-GMM	LDA-GMM
2	2	0.2468±0.05	0.3902±0.07	0.6692±0.03	0.6517±0.14	0.5576±0.11
	4	0.1897±0.01	0.4257±0.05	0.7261±0.03	0.6255±0.12	0.5350±0.09
	6	0.1970±0.02	0.4533±0.05	0.6510±0.09	0.6159±0.13	-
	9	0.2073±0.04	0.4902±0.06	0.6418±0.02	0.5899±0.09	-
3	2	0.2540±0.02	0.3900±0.06	0.6892±0.02	0.6032±0.15	0.5479±0.08
	4	0.2359±0.02	0.4164±0.02	0.6713±0.03	0.6076±0.10	0.5053±0.09
	6	0.2371±0.01	0.4615±0.04	0.6442±0.06	0.6088±0.08	-
	9	0.2085±0.04	0.5457±0.04	0.6088±0.07	0.6573±0.06	-
4	2	0.2530±0.05	0.3616±0.07	0.6986±0.05	0.6043±0.15	0.5279±0.09
	4	0.2528±0.05	0.4164±0.02	0.6345±0.04	0.6182±0.09	0.5550±0.06
	6	0.2560±0.03	0.4995±0.02	0.6219±0.06	0.6585±0.06	-
	9	0.2254±0.02	0.5553±0.05	0.6618±0.04	0.6964±0.03	-
5	2	0.2528±0.03	0.3892±0.06	0.6870±0.03	0.6149±0.12	0.5252±0.08
	4	0.2454±0.03	0.4459±0.04	0.6310±0.03	0.5887±0.09	0.4961±0.08
	6	0.2454±0.01	0.5362±0.04	0.6406±0.02	0.6973±0.07	-
	9	0.2169±0.04	0.5553±0.05	0.6406±0.02	0.6677±0.05	-

component number between 2 and 5. It is worth noting, that there are at most 21 samples for each class, which is less than the 39 dimensions.

Table 8 reports the results, where the proposed method outperforms the others. The performance difference is more prominent as the number of components and dimensions increases. Fig.8 summarizes some errors.

7. Conclusions and future work

In this paper, we have presented a novel joint learning model, referred to as 2L-MJFA, for classification. The model is very different from previous ap-

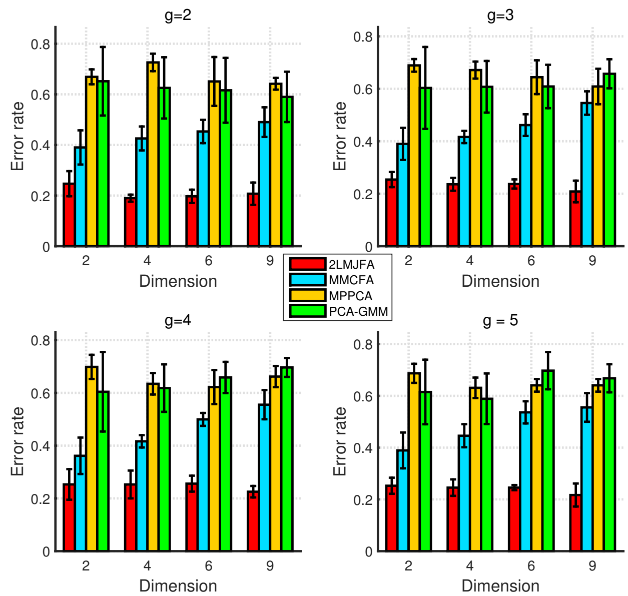


Figure 8: Error rate comparison for the BT dataset.

proaches, where dimensionality reduction is usually independent from the subsequent classification procedure. Specifically, it is based on a two-layer mixture or a mixture of mixtures structure, with each component that represents each specific class serving as another mixture model of factor analyzers designed to share the same loading matrix. The latter has a dual role with respect to being considered a dimensionality reduction matrix, and being capable for reducing the model parameters, making therefore the proposed algorithm very suitable for S3 problems. We have also described a modified EM algorithm to train the proposed model. A series of experiments has demonstrated that 2L-MJFA significantly outperforms three competitive algorithms on seven datasets. Future work includes exploring the possibility of determining the number of components and the dimensionality automatically via Bayesian learning type methodologies.

References

[1] S. Lacoste-Julien, F. Sha, M. I. Jordan, Discriminative learning for dimensionality reduction and classification, in: D. Koller, D. Schuurmans, Y. Bengio, L. Bottou (Eds.), Advances in Neural Information Processing Systems, (2009), pp. 897–904.

[2] G. Hinton, P. Dayan, M. Revow, Modeling the manifolds of images of handwritten digits, IEEE Transactions on Neural Networks 8 (1997) 65–74.

[3] G. J. McLachlan, D. Peel, Mixtures of factor analyzers, in: International Conference on Machine Learning (ICML), (2000), pp. 599–606.

[4] B. Xu, K. Huang, C.-L. Liu, Maxi-min discriminant analysis via online learning, Neural Networks 34 (2012) 56–64.

[5] K. Huang, D. Zheng, J. Sun, Y. Hotta, K. Fujimoto, S. Naoi, Sparse learning for support vector classification, Pattern Recognition Letters 31 (13) (2010) 1944–1951.

[6] K. Huang, I. King, M. R. Lyu, Direct zero-norm optimization for feature selection, in: Proceedings of the 8th IEEE International Conference on Data Mining, (2008), pp. 845–850.

[7] Z. K. Malik, A. Hussain, Q. M. J. Wu, An online generalized eigenvalue version of laplacian eigenmaps for visual big data, Neurocomputing 173 (2016) 127–136.

[8] A. Gisbrecht, A. Schulz, B. Hammer, Parametric nonlinear dimensionality reduction using kernel t-sne, Neurocomputing 147 (2015) 71–82.

[9] J. Venna, J. Peltonen, K. Nybo, H. Aidos, S. Kaski, Information retrieval perspective to nonlinear dimensionality reduction for data visualization, Journal of Machine Learning Research 11 (2010) 451–490.

[10] K. Huang, H. Yang, I. King, M. R. Lyu, Machine Learning: Modeling Data Locally and Gloablly, Springer Verlag, ISBN 3-5407-9451-4, (2008).

[11] X. Yang, K. Huang, Y. Goulermas, R. Zhang, Learning of unsupervised dimensionality reduction and gaussian mixture model, Neural Processing Letters (2016) (online).

[12] S. J. Raudys, A. K. Jain, Small sample size effects in statistical pattern recognition: Recommendations for practitioners, IEEE Transactions on Pattern Analysis and Machine Intelligence 13(3) (1991) 252–264.

[13] R. Huang, Q. Liu, H. Lu, S. Ma, Solving the small sample size problem of LDA, in: International Conference on Pattern Recognition (ICPR), (2002), pp. 29–32.

[14] J. Baek, G. J. McLachlan, L. K. Flack, Mixtures of factor analyzers with common factor loadings: Applications to the clustering and visualization of high-dimensional data, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (7) (2010) 1298–1309.

[15] M. E. Tipping, C. M. Bishop, Mixtures of probabilistic principal component analysers, Neural Computation 11(2) (2006) 443–482.

[16] M. E. Tipping, C. M. Bishop, Probabilistic principal component analysis, Journal of the Royal Statistical Society, Series B 61 (1999) 611–622.

[17] X. Yang, K. Huang, R. Zhang, J. Y. Goulermas, Two-layer mixture of factor analyzers with joint factor loading, in: International Joint Conference on Neural Networks, (2015), pp. 1–8.

[18] X. Wei, C. Li, Bayesian mixtures of common factor analyzers: Model, variational inference, and applications, Signal Processing 93 (11) (2013) 2894–2905.

[19] W. Wang, Mixtures of common factor analyzers for high-dimensional data with missing information, J. Multivariate Analysis 117 (2013) 120–133.

[20] A. Basilevsky, Statistical Factor Analysis and Related Methods, New York: Wiley, (1994).

[21] G. J.Mclanchlan, D. Peel, R. W. Bean, Modelling high-dimensional data by mixtures of factor analyzers, Computational Statistics & Data Analysis 41 (2003) 379–388.

[22] Z. Ghahramani, G. Hinton, The em algorithm for mixtures of factor analyzers, in: Technical Report CRG-TR-96-1, University of Toronto, http://www.gatsby.ucl.ac.uk/.zoubin/papers.html, (1996), pp. 11–18.

[23] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the Royal Statistical Society, Series B 39 (1) (1977) 1–38.

[24] G. McLachlan, T. Krishnan, The EM algorithm and extensions, Vol. 382, John Wiley & Sons, (2007).

[25] A. Montanari, C. Viroli, Maximum likelihood estimation of mixtures of factor analyzers, Computational Statistics & Data Analysis 55 (9) (2011) 2712–2723.

[26] M. Kearns, Y. Mansour, A. Y. Ng, An information-theoretic analysis of hard and soft assignment methods for clustering, CoRR abs/1302.1552.

[27] H. T. Kahraman, S. Sagiroglu, I. Colak, Developing intuitive knowledge classifier and modeling of users' domain dependent data in web, Knowledge Based Systems 37 (2013) 283–295.

[28] W. Street, W. Wolberg, O. Mangasarian, Nuclear feature extraction for breast tumor diagnosis, IST/SPIE 1993 International Symposium on Electronic Imaging: Science and Technology 1905 (1993) 861–870.

[29] O. Mangasarian, W. Street, W. Wolberg, Breast cancer diagnosis and prognosis via linear programming, Operations Research 43(4) (1995) 26–30.

[30] B. Johnson, Z. Xie, Classifying a high resolution image of an urban area using super-object information, ISPRS Journal of Photogrammetry and Remote Sensing 83 (2013) 40–49.

[31] B. Johnson, High resolution urban land cover classification using a competitive multi-scale object-based approach, Remote Sensing Letters 4 (2) (2013) 131–140.

[32] A. Tsanas, M. Little, C. Fox, L. Ramig, Objective automatic assessment of rehabilitative speech treatment in parkinson disease, IEEE Transactions on Neural Systems and Rehabilitation Engineering 22 (2014) 181–190.

[33] J. E. Silva, J. P. Marques de Sa, J. Jossinet, Classification of breast tissue by electrical impedance spectroscopy, Medical and Biological Engineering and Computing 38 (2000) 26–30.