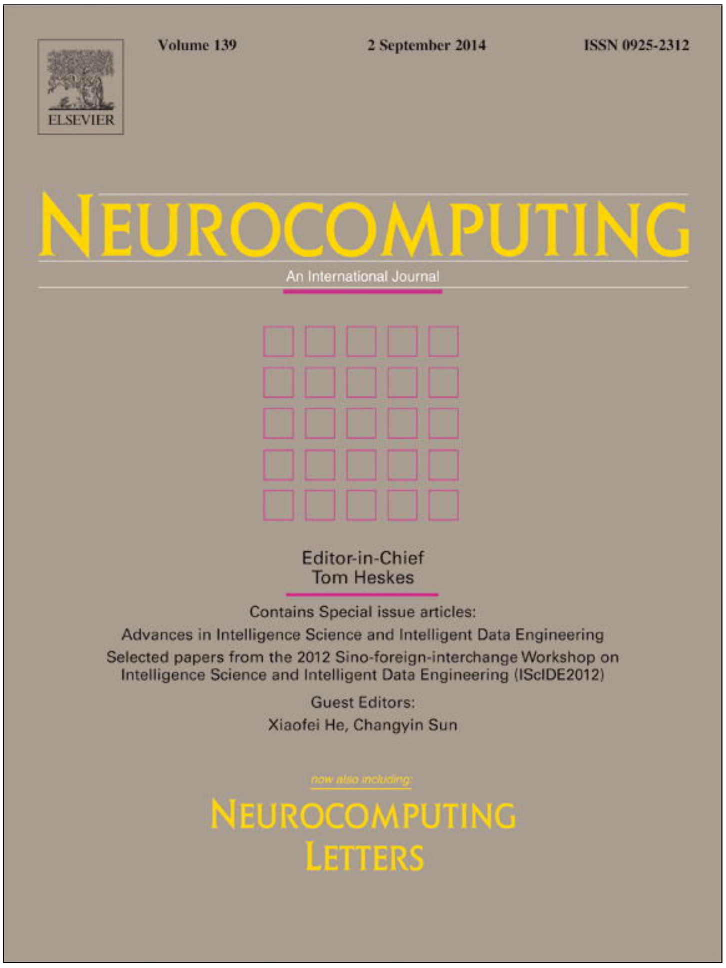


Volume 139 2 September 2014 ISSN 0925-2312

ELSEVIER

NEUROCOMPUTING
An International Journal

Editor-in-Chief
Tom Heskes

Contains Special issue articles:
Advances in Intelligence Science and Intelligent Data Engineering
Selected papers from the 2012 Sino-foreign-interchange Workshop on
Intelligence Science and Intelligent Data Engineering (IScIDE2012)

Guest Editors:
Xiaofei He, Changyin Sun

now also including:
NEUROCOMPUTING
LETTERS

Neurocomputing 139 (2014) 3–14

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Learning local factor analysis versus mixture of factor analyzers with automatic model selection

Lei Shi [a], Zhi-Yong Liu [b], Shikui Tu [a], Lei Xu [a,*]

[a] Department of Computer Science and Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[b] The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China

ARTICLE INFO

Article history:
Received 3 June 2013
Received in revised form
29 August 2013
Accepted 15 September 2013
Available online 31 March 2014

Keywords:
Automatic model selection
Mixture of factor analyzers
Local factor analysis
Variational Bayes
Bayesian Ying-Yang
Dirichlet–Normal–Gamma

ABSTRACT

Considering Factor Analysis (FA) for each component of Gaussian Mixture Model (GMM), clustering and local dimensionality reduction can be addressed simultaneously by Mixture of Factor Analyzers (MFA) and Local Factor Analysis (LFA), which correspond to two FA parameterizations, respectively. This paper investigates the performance of Variational Bayes (VB) and Bayesian Ying-Yang (BYY) harmony learning on MFA/LFA for the problem of automatically determining the component number and the local hidden dimensionalities (i.e., the number of factors of FA in each component). Similar to the existing VB learning algorithm on MFA, we develop an alternative VB algorithm on LFA with a similar conjugate Dirichlet–Normal–Gamma (DNG) prior on all parameters of LFA. Also, the corresponding BYY algorithms are developed for MFA and LFA. A wide range of synthetic experiments shows that LFA is superior to MFA in model selection under either VB or BYY, while BYY outperforms VB reliably on both MFA and LFA. These empirical findings are consistently observed from real applications on not only face and handwritten digit images clustering, but also unsupervised image segmentation.

1. Introduction

Mixture models [1,2], such as Gaussian Mixture Model (GMM) [3,4], have been widely used in many applications. By exploiting the Factor Analysis (FA) [5] in each Gaussian component, the correlated high dimensional data can be represented by fewer latent factors without requiring $\mathcal{O}(d^2)$ parameters for each Gaussian covariance matrix, where d is the dimensionality of the data. The mixture model can be regarded as a constrained GMM, and has been studied under the name of Mixture of Factor Analyzers (MFA) [2,6] or Local Factor Analysis (LFA) [7,8] in the literature. MFA and LFA separately employ two parameterizations of FA, shortly called as FA-a that takes the form of a free factor loading matrix and an identity covariance matrix for the latent factors, and FA-b that constrains the factor loading matrix to be a rectangular orthogonal matrix, and allows a diagonal covariance matrix for the latent variables, respectively in [9].

Learning MFA/LFA includes parameter learning for estimating all the unknown parameters and model selection for determining the component number k and the hidden dimensionalities $\{h_i\}_{i=1}^k$. Parameter learning is usually implemented under the maximum likelihood principle by an Expectation–Maximization (EM) algorithm [1,10,11]. A conventional model selection approach is featured by a two-stage implementation. The first stage conducts parameter learning for each $\mathbf{k} \in \mathcal{M}$ to get a set of candidate models, where $\mathbf{k} = \{k, \{h_i\}\}$ for MFA/LFA. The second stage selects the best candidate by a model selection criterion, e.g., Akaike's Information Criterion (AIC) [12]. However, this two-stage implementation suffers from a huge computation because it requires parameter learning for each $\mathbf{k} \in \mathcal{M}$. Moreover, a larger $\mathbf{k}$ often implies more unknown parameters, and then parameter estimation becomes less reliable so that the criterion evaluation reduces its accuracy (see Section 2.1 in [13] for a detailed discussion).

To reduce the computation, an Incremental Mixture of Factor Analyzers (IMoFA) algorithm was proposed on MFA in [14] with the validation likelihood as the criterion to judge whether to split a component, or add a hidden dimension, or terminate. Although such an incremental procedure can save the costs to some extent, it usually leads to a suboptimal solution [13,15].

Another road is referred to as automatic model selection, which starts from a large enough $\mathbf{k}$, and has an intrinsic force to drive extra structures diminished, and thus automatically determines $\mathbf{k}$ during parameter learning. An early effort is Rival Penalized Competitive Learning (RPCL) on GMM [16,17]. Two Bayesian related approaches can be implemented with a nature of automatic model selection. One is Bayesian Ying-Yang (BYY) learning,

* Corresponding author.
E-mail address: lxu@cse.cuhk.edu.hk (L. Xu).

http://dx.doi.org/10.1016/j.neucom.2013.09.061
0925-2312/© 2014 Elsevier B.V. All rights reserved.

proposed in [18] and systematically developed in the past decade and a half [13,15,19,20], which provides a general statistical learning framework that can handle both parameter learning and model selection under a best harmony principle. BYY is capable of automatic model selection even without imposing any priors on the parameters, and its performance can be further improved with appropriate priors incorporated according to a general guideline. The other is Variational Bayes (VB) [6,21]. It tackles the difficulty in computing the marginal likelihood with a lower bound by means of variational method, and an EM-like algorithm is employed to optimize this lower bound. The model selection of VB is realized by incorporating an appropriate prior distributions on the parameters.

Recently, a comparative study [4] was delivered on automatic model selection by BYY, VB and MML (Minimum Message Length) for GMM with priors over the parameters. Also in [9], FA-b shows better model selection performance than FA-a under BYY and VB, although FA-a and FA-b have equivalent likelihood functions.

This paper is motivated for an empirical investigation on the automatic model selection performances of BYY and VB, based on MFA and LFA, which actually correspond to Mixture of FA-a and Mixture of FA-b, respectively. There exists a VB algorithm [6] for MFA with a Dirichlet prior on the mixing weights, Normal priors on the columns of the factor-loading matrix, and Gamma priors on precision parameters. Following [4], we consider a full prior on all parameters and adopt a Normal prior over the mean vector in each component of MFA. For short, DNG is referred to the above Dirichlet, Normal, Gamma priors. By slightly modifying the one in [6], we obtain a VB learning algorithm with the DNG prior, shortly denoted as VB-MFA. Also, a similar conjugate DNG prior is considered on the parameters of LFA.

Moreover, we develop three automatic model selection algorithms, namely the VB algorithm on LFA, or VB-LFA for short, and the BYY algorithms on MFA and LFA, shortly denoted as BYY-MFA and BYY-LFA respectively. With the conjugate property of the priors, the BYY harmony measure is computed by directly integrating out the parameters with respect to the Yang posteriors, instead of using Taylor approximations as in [9]. The handled marginal density of observed variable in each component is tackled by a lower-bound approximation with the help of additional variables, leading to products of multiple Student's T-distributions.

The performances of automatic model selection are extensively compared on a wide range of randomly simulated data, via controlling the hardness of tasks by varying the dimension of data, the number of samples, the true number of components, and the overlap degree of components. The simulated results show the following empirical findings. First, LFA gets better performance than MFA under either VB or BYY, which echoes the advantages of FA-b over FA-a observed in [9]. Second, BYY outperforms VB on both MFA and LFA, and in most cases BYY-LFA performs the best. Also, we apply these algorithms to not only clustering face and handwritten digit images, but also unsupervised image segmentation on real world images. The results are consistent with the observations from simulated experiments.

The main contribution of this paper can be summarized in two-fold. First, three algorithms, i.e, the algorithm of VB based LFA with Dirichlet–Normal–Gamma (DNG) prior, denoted by VB-LFA, the algorithm of BYY based LFA with DNG prior, denoted by BYY-LFA, and the algorithm of BYY based MFA with DNG prior, denoted by BYY-MFA are derived in detail. Second, based on the algorithms, we empirically compared by extensive experiments the two types of clustering of factor analysis models, i.e., LFA and MFA, as well as two types of automatic model selection strategies, i.e., VB and BYY.

The remainder of this paper is organized as follows. Section 2 introduces MFA/LFA and their DNG priors. We introduce the automatic model selection algorithms with the DNG priors by BYY in Section 3, and by VB in Section 4. Experimental comparisons are conducted via a wide range of synthetic datasets and real applications in Section 5. Finally, concluding remarks are made in Section 6.

2. Models and priors

2.1. Model parameterizations

In a mixture model, the distribution $q(\mathbf{x}|\boldsymbol{\Theta})$ of a d-dimensional observed random variable $\mathbf{x}$ is a mixture of several local distributions $q(\mathbf{x}|i,\boldsymbol{\theta})$, with each named as a component:

$$q(\mathbf{x}|\boldsymbol{\Theta}) = \sum_{i=1}^{k} \alpha_i q(\mathbf{x}|i,\boldsymbol{\theta}_i) \quad \text{with} \quad \boldsymbol{\Theta} = \{\alpha_i\} \cup \{\boldsymbol{\theta}_i\}, \tag{1}$$

where k is the component number, $\{\alpha_i\}$ are mixing weights with $\sum_{i=1}^{k}\alpha_i = 1$ and each $\alpha_i \geq 0$, and $\boldsymbol{\theta}_i$ denotes parameters of the ith component. Here and throughout this paper, $q(\cdot)$ is referred to as a generative distribution, likelihood or prior, while $p(\cdot)$ is referred to as a posterior distribution.

If each component is a Gaussian distribution, i.e., $q(\mathbf{x}|i,\boldsymbol{\Theta}) = G(\mathbf{x}|\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_{x|i})$ with mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_{x|i}$, $q(\mathbf{x}|\boldsymbol{\Theta})$ by Eq. (1) becomes the widely used Gaussian Mixture Model. For a full matrix $\boldsymbol{\Sigma}_{x|i}$, there are $0.5d(d+1)$ free parameters to be estimated, whose accuracy is difficult be guaranteed for a small sample size. One way for tackling this problem is to impose certain constraints on $\boldsymbol{\Sigma}_{x|i}$ with a Factor Analysis model, i.e.,

$$\begin{aligned} &q(\mathbf{x}|\mathbf{y},i,\boldsymbol{\theta}_i) = G(\mathbf{x}|\mathbf{A}_i\mathbf{y}+\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i), \quad q(\mathbf{y}|i,\boldsymbol{\theta}_i) = G(\mathbf{y}|\mathbf{0},\boldsymbol{\Sigma}_{y|i}),\\ &q(\mathbf{x}|i,\boldsymbol{\theta}_i) = \int q(\mathbf{x}|\mathbf{y},i,\boldsymbol{\theta}_i)q(\mathbf{y}|i,\boldsymbol{\theta}_i)\,d\mathbf{y} = G(\mathbf{x}|\boldsymbol{\mu}_i,\mathbf{A}_i\boldsymbol{\Sigma}_{y|i}\mathbf{A}_i^T+\boldsymbol{\Psi}_i), \end{aligned} \tag{2}$$

where we introduce a hidden factor $\mathbf{y}$ in an h_i-dimensional subspace with $h_i < d$, and constrain $\boldsymbol{\Psi}_i$ to be diagonal. FA actually factorizes $\boldsymbol{\Sigma}_{x|i}$ to be $\boldsymbol{\Sigma}_{x|i} = \mathbf{A}_i\boldsymbol{\Sigma}_{y|i}\mathbf{A}_i^T+\boldsymbol{\Psi}_i$ with fewer free parameters.

To reduce the indeterminacies of the FA by Eq. (2), two parameterizations of FA are typically used, called as Mixture of Factor Analyzers (MFA) [2,6] and Local Factor Analysis (LFA) [7,8] respectively, with their corresponding mixture models by Eq. (1) summarized in Table 1. The two FA parameterizations have equivalent likelihood functions by Eq. (2), and thus they have the same model selection performance in a two-stage implementation with AIC or BIC [22]. However, it was found that they result in different model selection performances under BYY [23], and a recent study [9] provided systematic empirical findings on how parameterizations affect model selection performance under not only BYY but also VB. Moreover, the differences of two parameterizations on model selection performance have been further analytically investigated in Section 2.2 of [20]. In this paper, we proceed to investigate the automatic model selection performances of MFA/LFA under BYY and VB.

Moreover, when each diagonal covariance $\boldsymbol{\Psi}_i$ in Table 1 is constrained to be spherical, i.e., $\boldsymbol{\Psi}_i = \psi_i\mathbf{I}_d$, MFA and LFA will degenerate to Mixture of PCA [11] and Local PCA [8], respectively.

Table 1
MFA v.s. LFA: similarity and difference. MFA and LFA are actually mixtures of FA-a and FA-b in [9], respectively.

Model:	MFA (mixture of FA-a)	LFA (mixture of FA-b)
Parameters $\boldsymbol{\theta}_i$:	$\{\mathbf{A}_i,\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i\}$	$\{\mathbf{U}_i,\boldsymbol{\Lambda}_i,\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i\}$
Same:	$\boldsymbol{\Psi}_i$ is $d \times d$ diagonal	$\boldsymbol{\Psi}_i$ is $d \times d$ diagonal
Different:	$\mathbf{A}_i$ is general $d \times h_i$	$\mathbf{U}_i$ is orthogonal, i.e., $\mathbf{U}_i^T\mathbf{U}_i = \mathbf{I}_{h_i}$, $\boldsymbol{\Lambda}_i$ is diagonal, $\boldsymbol{\Lambda}_i = \mathrm{diag}[\lambda_1,\ldots,\lambda_{h_i}]$
$q(\mathbf{y}\|i,\boldsymbol{\theta}_i)$:	$G(\mathbf{y}\|\mathbf{0},\mathbf{I}_{h_i})$	$G(\mathbf{y}\|\mathbf{0},\boldsymbol{\Lambda}_i)$
$q(\mathbf{x}\|\mathbf{y},i,\boldsymbol{\theta}_i)$:	$G(\mathbf{x}\|\mathbf{A}_i\mathbf{y}+\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i)$	$G(\mathbf{x}\|\mathbf{U}_i\mathbf{y}+\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i)$
$q(\mathbf{x}\|i,\boldsymbol{\theta}_i)$:	$G(\mathbf{x}\|\boldsymbol{\mu}_i,\mathbf{A}_i\mathbf{A}_i^T+\boldsymbol{\Psi}_i)$	$G(\mathbf{x}\|\boldsymbol{\mu}_i,\mathbf{U}_i\boldsymbol{\Lambda}_i\mathbf{U}_i^T+\boldsymbol{\Psi}_i)$

Additionally, MFA/LFA can be reformulated by introducing a binary variable set $\mathbf{Z}=\{\mathbf{z}_t\}_{t=1}^{N}$ corresponding to the i.i.d. samples $\mathbf{X}_N=\{\mathbf{x}_t\}_{t=1}^{N}$. In each $\mathbf{z}_t$, we have each ith element $z_{it}\in\{0,1\}$ and $\sum_{i=1}^{k} z_{it}=1$, and $z_{it}=1$ iff $\mathbf{x}_t$ is generated from the ith component. The generative process of an observation $\mathbf{x}_t$ is thus interpreted as three steps: (1) sample $\mathbf{z}_t$ from a Multinomial distribution described by $\boldsymbol{\alpha}=[\alpha_1,\dots,\alpha_k]^T$, i.e., $\mathbf{z}_t\sim Multinomial(\boldsymbol{\alpha})$; (2) generate hidden variable $\mathbf{y}$ from a Gaussian subspace, i.e., $\mathbf{y}\sim\prod_i q(\mathbf{y}|i,\boldsymbol{\theta}_i)^{z_{it}}$; (3) generate $\mathbf{x}_t$ from a Gaussian conditional on $\mathbf{y}$, i.e., $\mathbf{x}_t\sim\prod_i q(\mathbf{x}_t|\mathbf{y},i,\boldsymbol{\theta}_i)^{z_{it}}$. Therefore, we have the following joint probability:

$$q(\mathbf{X}_N,\mathbf{Y},\mathbf{Z}|\boldsymbol{\Theta})=\prod_{t=1}^{N}\prod_{i=1}^{k}[\alpha_i q(\mathbf{x}_t|\mathbf{y},i,\boldsymbol{\theta}_i)q(\mathbf{y}|i,\boldsymbol{\theta}_i)]^{z_{it}}. \tag{3}$$

Given a set of observations $\mathbf{X}_N$, supposing that the component number k and the local hidden dimensionalities $\{h_i\}$ are given, one widely used method for parameter estimation is the maximum-likelihood learning, which can be effectively implemented by the well-known Expectation–Maximization (EM) algorithm [1]. Model selection on LFA/MFA is to appropriately determine both the component number k and the local hidden dimensionalities $\{h_i\}_{i=1}^{k}$, or shortly to determine the tuple $\mathbf{k}=\{k,\{h_i\}_{i=1}^{k}\}$, on which the maximum likelihood principle fails to give a good guide [13].

2.2. The conjugate Dirichlet–Normal–Gamma priors

Considering the parameters with appropriate prior distributions can provide helpful learning regularization and also improve model selection performance [13]. Such empirical studies have been conducted on GMM with either the improper Jeffreys prior or the conjugate Dirichlet–Normal–Wishart prior under BYY, VB and MML [4], and also on FA with Normal–Gamma prior under BYY and VB [9]. This section considers similar prior distributions on parameters $\boldsymbol{\Theta}=\boldsymbol{\alpha}\cup\{\boldsymbol{\theta}_i\}_{i=1}^{k}$ of MFA or LFA in Table 1, where the Dirichelet distribution $D(\boldsymbol{\alpha}|\boldsymbol{\lambda},\xi)$ takes the form,

$$D(\boldsymbol{\alpha}|\boldsymbol{\lambda},\xi)=\frac{\Gamma(\xi)}{\prod_{i=1}^{k}\Gamma(\xi\lambda_i)}\left(\prod_{i=1}^{k}\alpha_i^{\xi\lambda_i-1}\right), \tag{4}$$

with the constraints $\boldsymbol{\lambda}=[\lambda_1,\dots,\lambda_k]^T$, $\sum_i\lambda_i=1$, $\forall\lambda_i\geq 0$.

For MFA, we consider a Dirichlet prior on the mixing weights $\boldsymbol{\alpha}=[\alpha_1,\dots,\alpha_k]^T$, a Normal prior on each component's mean vector $\boldsymbol{\mu}_i$, an independent Gamma prior on the diagonal elements of $\boldsymbol{\varphi}_i=\boldsymbol{\Psi}_i^{-1}$, where $\varphi_i^{(j)}$ is the jth diagonal element in $\boldsymbol{\varphi}_i$. Moreover, a hierarchical Normal–Gamma prior is assigned on each jth column $\mathbf{A}_i^{(*j)}$ of $\mathbf{A}_i$, where $\mathbf{A}_i^{(*j)}$ *a priori* comes from a zero-mean Normal distribution with a covariance $\mathbf{I}_d/\varsigma_{ij}$, and ς_{ij} further follows a Gamma prior. We use $\Gamma(\cdot|a,b)$ to denote the Gamma distribution with a shape parameter $a>0$ and an inverse scale parameter $b>0$. The whole $q(\boldsymbol{\Theta})$ is shortly denoted as DNG with details given in the left of Table 2. A DNG prior was considered on MFA under VB in [6] without $q(\boldsymbol{\mu}_i)$. Based on the observations in [4] that a full prior helps to improve model selection performance, in this paper we consider the full DNG prior with $q(\boldsymbol{\mu}_i)$.

Table 2
The Dirichlet–Normal–Gamma priors on MFA and LFA with hyper-parameters $\boldsymbol{\Xi}=\{\boldsymbol{\lambda},\xi,\beta\}\cup\{\boldsymbol{\Xi}_i\}_{i=1}^{k}$. Priors on MFA/LFA are described in the two big columns respectively, whose differences are highlighted with gray color. The "distr." columns indicate the distribution types for clarity, with "D" for Dirichlet, "N" for Normal, and "G" for Gamma, respectively.

MFA		LFA	
$\boldsymbol{\Theta}=\boldsymbol{\alpha}\cup\{\boldsymbol{\theta}_i\}_i$, $\boldsymbol{\theta}_i=\{\mathbf{A}_i,\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i\}$		$\boldsymbol{\Theta}=\boldsymbol{\alpha}\cup\{\boldsymbol{\theta}_i\}_i$, $\boldsymbol{\theta}_i=\{\mathbf{U}_i,\boldsymbol{\Lambda}_i,\boldsymbol{\mu}_i,\boldsymbol{\Psi}_i\}$	
$\boldsymbol{\varphi}_i=\boldsymbol{\Psi}_i^{-1}$; $\mathbf{A}_i^{(*j)}$: jth column vector of $\mathbf{A}_i$		$\boldsymbol{\nu}_i=\boldsymbol{\Lambda}_i^{-1}$, $\boldsymbol{\varphi}_i=\boldsymbol{\Psi}_i^{-1}$	
$\boldsymbol{\Xi}_i=\{\mathbf{m}_i,\{a_{ij}^{\varphi},b_{ij}^{\varphi}\}_j,\{a_{ij}^{\varsigma},b_{ij}^{\varsigma}\}_j\}$		$\boldsymbol{\Xi}_i=\{\mathbf{m}_i,\{a_{ij}^{\varphi},b_{ij}^{\varphi}\}_j,\{a_{ij}^{\nu},b_{ij}^{\nu}\}_j\}$	
prior	distr.	prior	distr.
$q(\boldsymbol{\Theta})=q(\boldsymbol{\alpha})\prod_{i=1}^{k}q(\boldsymbol{\theta}_i)$	DNG	$q(\boldsymbol{\Theta})=q(\boldsymbol{\alpha})\prod_{i=1}^{k}q(\boldsymbol{\theta}_i)$	DNG
$q(\boldsymbol{\alpha})=\mathcal{D}(\boldsymbol{\alpha}\|\boldsymbol{\lambda},\xi)$	D	$q(\boldsymbol{\alpha})=\mathcal{D}(\boldsymbol{\alpha}\|\boldsymbol{\lambda},\xi)$	D
$q(\boldsymbol{\theta}_i)=q(\boldsymbol{\mu}_i)q(\boldsymbol{\varphi}_i)q(\mathbf{A}_i)$	NG	$q(\boldsymbol{\theta}_i)=q(\boldsymbol{\mu}_i)q(\boldsymbol{\varphi}_i)q(\boldsymbol{\nu}_i)$	NG
$q(\boldsymbol{\mu}_i)=G(\boldsymbol{\mu}_i\|\mathbf{m}_i,\mathbf{I}_d/\beta)$	N	$q(\boldsymbol{\mu}_i)=G(\boldsymbol{\mu}_i\|\mathbf{m}_i,\mathbf{I}_d/\beta)$	N
$q(\boldsymbol{\varphi}_i)=\prod_{j=1}^{d}\Gamma(\varphi_i^{(j)}a_{ij}^{\varphi},b_{ij}^{\varphi})$	G	$q(\boldsymbol{\varphi}_i)=\prod_{j=1}^{d}\Gamma(\varphi_i^{(j)}\|a_{ij}^{\varphi},b_{ij}^{\varphi})$	G
$q(\mathbf{A}_i\|\boldsymbol{\varsigma}_i)=\prod_{j=1}^{h_i}G(\mathbf{A}_i^{(*j)}\|\mathbf{0},\mathbf{I}_d/\varsigma_i^{(j)})$	N	$q(\boldsymbol{\nu}_i)=\prod_{j=1}^{h_i}\Gamma(\nu_i^{(j)}\|a_{ij}^{\nu},b_{ij}^{\nu})$	G
$q(\boldsymbol{\varsigma}_i)=\prod_{j=1}^{h_i}\Gamma(\varsigma_i^{(j)}\|a_{ij}^{\varsigma},b_{ij}^{\varsigma})$	G		

For LFA, we consider a similar DNG prior on $\boldsymbol{\Theta}$ in the right of Table 2, with the parts different from MFA highlighted by gray color. Therein, each orthogonal matrix $\mathbf{U}_i$ is considered without any prior, instead of adopting the $q(\mathbf{U}_i)$ used in [9] because it is irrelevant to $\mathbf{U}_i$ and thus not helpful for automatic model selection. The differences in $q(\boldsymbol{\Theta})$ actually come from the parameterizations. Therefore, the $q(\boldsymbol{\Theta})$ for LFA is also called DNG prior in this paper without ambiguity. For both MFA and LFA, the DNG prior is conjugate [6,11] to the generative process described in Eq. (3).

In the sequel, we may use short notations $\mathbf{a}_i^{\varphi}=[a_{i1}^{\varphi},\dots,a_{id}^{\varphi}]^T$, $\mathbf{b}_i^{\varphi}=[b_{i1}^{\varphi},\dots,b_{id}^{\varphi}]^T$, $\mathbf{a}_i^{\varsigma}=[a_{i1}^{\varsigma},\dots,a_{ih_i}^{\varsigma}]^T$, $\mathbf{b}_i^{\varsigma}=[b_{i1}^{\varsigma},\dots,b_{ih_i}^{\varsigma}]^T$, $\mathbf{a}_i^{\nu}=[a_{i1}^{\nu},\dots,a_{ih_i}^{\nu}]^T$ and $\mathbf{b}_i^{\nu}=[b_{i1}^{\nu},\dots,b_{ih_i}^{\nu}]^T$ for expression convenience.

3. BYY algorithms for learning LFA versus MFA

3.1. Bayesian Ying-Yang (BYY) harmony learning

Firstly proposed in [18] and systematically developed over a decade and a half [13], Bayesian Ying-Yang (BYY) harmony learning theory is a general statistical learning framework that can handle both parameter learning and model selection under a best harmony principle, which provides a favorable new mechanism for model selection.

The BYY harmony learning is featured by seeking the best harmony between the Ying- Yang pair in a BYY system. The BYY system consists of Yang machine and Ying machine, respectively corresponding to two types of decompositions, namely Yang $p(\mathbf{R}|\mathbf{X})p(\mathbf{X})$ and Ying $q(\mathbf{X}|\mathbf{R})q(\mathbf{R})$, where the observed data $\mathbf{X}$ is regarded as generated from its inner representation $\mathbf{R}=\{\mathbf{Y},\boldsymbol{\Theta}\}$ that consists of latent variables $\mathbf{Y}$ and parameters $\boldsymbol{\Theta}$, supported by a hyper-parameter set $\boldsymbol{\Xi}$. The harmony measure is mathematically expressed as follows [13,15,19]:

$$H(p||q,\boldsymbol{\Xi})=\int p(\mathbf{R}|\mathbf{X})p(\mathbf{X})[q(\mathbf{X}|\mathbf{R})q(\mathbf{R})]d\mathbf{X}\,d\mathbf{R}. \tag{5}$$

Maximizing $H(p||q,\boldsymbol{\Xi})$ leads to not only a best matching between the Ying-Yang pair, but also a compact model with a least complexity. Different from VB model selection that bases on an appropriate prior $q(\boldsymbol{\Theta}|\boldsymbol{\Xi})$ (see Section 4 for the details of the prior in VB), BYY harmony learning by Eq. (5) bases on $q(\mathbf{R})=q(\mathbf{Y}|\boldsymbol{\Theta})q(\boldsymbol{\Theta}|\boldsymbol{\Xi})$ to make model selection, where $q(\mathbf{Y}|\boldsymbol{\Theta})$ plays a role that is not only equally important to $q(\boldsymbol{\Theta}|\boldsymbol{\Xi})$ but also easy computing, and $q(\boldsymbol{\Theta}|\boldsymbol{\Xi})$ is still handled in a way similar to VB. Moreover, maximizing $H(p||q,\boldsymbol{\Xi})$ is implemented with the help of the general two-stage iterative procedure shown by Fig. 6 in [19] (also see Eqs. (6) and (7) in [8]). The first stage estimates $\boldsymbol{\Xi}$ (usually via estimating $\boldsymbol{\Theta}$) by an optimization of continuous variables, while the second stage involves a discrete optimization on one or several integers that index candidate models. Here, we only consider the first stage where automatic model selection actually performs, though the second stage may be also considered to further improve the model selection performance with much more computing costs.

BYY harmony learning leads to improved model selection via either or both of improved model selection criteria and algorithms with automatic model selection. Such a merit can be intuitively understood as follows [13]:

$$H(p||q) = \int p(\mathbf{X})\ln q(\mathbf{X})\, d\mathbf{X} = H(p||p) - KL(p||q). \tag{6}$$

Thus, besides the Kullback–Leibler divergence, a system entropy term $H(p||p)$ is also incorporated into the BYY objective function. By contrast, VB tries to maximize the marginal likelihood by minimizing only the Kullback–Leibler divergence. It is the term $H(p||p)$ that makes the BYY harmony learning possess automatic model selection ability, even there is no prior on the parameter. More specifically, to estimate some mixture model such as the MFA, with the help of $H(p||p)$ the BYY harmony learning takes the following E-step which compared with the conventional EM algorithm has an extra term $\varDelta$ as follows [13]:

$$p_{j,t} = p(j|t) + \varDelta_{j,t}. \tag{7}$$

By such a regularization term $\varDelta_{j,t}$ (see also (A.9) in Appendix A), the updating on the jth component shares somewhat a similar updating to the rival penalized competitive learning (RPCL) [16,17], and thus realizes automatic model selection.

On MFA and LFA with the DNG priors in a conjugate manner, there is still no detailed automatic model selection algorithm available for implementing BYY harmony learning, and thus this section targets at developing such algorithms.

3.2. BYY algorithm for learning LFA with DNG prior

For LFA, we consider the Ying machine as $q(\mathbf{X},\mathbf{R}) = q(\mathbf{X},\mathbf{Y},\mathbf{Z}|\mathbf{\Theta})q(\mathbf{\Theta})$, with $\mathbf{R} = \{\mathbf{Y},\mathbf{Z},\mathbf{\Theta}\}$, $q(\mathbf{X},\mathbf{Y},\mathbf{Z}|\mathbf{\Theta})$ given in Eq. (3) and $q(\mathbf{\Theta})$ given in the right of Table 2. In the Yang machine, we consider $p(\mathbf{X})$ as the empirical distribution, i.e., $p(\mathbf{X}) = \delta(\mathbf{X}-\mathbf{X}_N)$, and the Yang-pathway $p(\mathbf{R}|\mathbf{X})$ as

$$\begin{aligned}
p(\mathbf{R}|\mathbf{X}_N) &= p(\mathbf{\Theta}|\mathbf{Z},\mathbf{Y},\mathbf{X}_N)p(\mathbf{Y}|\mathbf{Z},\mathbf{X}_N)p(\mathbf{Z}|\mathbf{X}_N),\\
p(\mathbf{\Theta}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N) &= p(\boldsymbol{\alpha}|\mathbf{Z},\mathbf{X}_N)\prod_{i=1}^{k}\left[p(\boldsymbol{\mu}_i|\mathbf{Z},\mathbf{X}_N)\prod_{j=1}^{h_i}p(\nu_i^{(j)}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N)\right.\\
&\qquad\left.\times\prod_{j=1}^{d}p(\varphi_i^{(j)}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N)\right],\\
p(\mathbf{Y}|\mathbf{Z},\mathbf{X}_N) &= \prod_{i=1}^{k}\prod_{t=1}^{N}p(\mathbf{y}|i,\mathbf{x}_t)^{z_{it}},\\
p(\mathbf{Z}|\mathbf{X}_N) &= \prod_{i=1}^{k}\prod_{t=1}^{N}p(i|\mathbf{x}_t)^{z_{it}}.
\end{aligned}\tag{8}$$

Particularly, in accordance with the variety preservation principle (see Section 4.2 in [13]), the details of $p(\mathbf{\Theta}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N)$ are further designed as the following posteriors in the DNG form by utilizing the conjugate property:

$$\begin{aligned}
p(\boldsymbol{\alpha}|\mathbf{Z},\mathbf{X}_N) &= \mathcal{D}(\boldsymbol{\alpha}|\lambda^*,\xi+N),\\
p(\boldsymbol{\mu}_i|\mathbf{Z},\mathbf{X}_N) &= G\left(\boldsymbol{\mu}_i|\mathbf{m}_i^*,\left[\beta\mathbf{I}_d+\left(\sum_{t=1}^{N}z_{it}\right)\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})\right]^{-1}\right),\\
p(\nu_i^{(j)}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N) &= \Gamma\left(\nu_i^{(j)}|a_{ij}^{\nu}+\frac{1}{2}\sum_{t=1}^{N}z_{it},b_{ij}^{*\nu}\right),\\
p(\varphi_i^{(j)}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N) &= \Gamma\left(\varphi_i^{(j)}|a_{ij}^{\varphi}+\frac{1}{2}\sum_{t=1}^{N}z_{it},b_{ij}^{*\varphi}\right),
\end{aligned}\tag{9}$$

where $\{\lambda^*,\mathbf{m}_i^*,b_{ij}^{*\nu},b_{ij}^{*\varphi}\}$ are free hyper-parameters to be optimized. Therein and throughout this paper, we use symbols "$\oslash$" and "$\odot$" to denote the Hadamard (element-by-element) division and product respectively.

The $p(i|\mathbf{x}_t)$ in Eq. (8) is constructed as the Bayesian posterior of $q(\mathbf{x}_t,i)$, i.e., $p(i|\mathbf{x}_t) \propto q(\mathbf{x}_t,i)$, where the $q(\mathbf{x}_t,i)$ is computed by

$$q(\mathbf{x}_t,i) = \int q(\mathbf{x}_t,i|\mathbf{\Theta})q(\mathbf{\Theta})\,d\mathbf{\Theta} = \lambda_i q(\mathbf{x}_t|i), \tag{10}$$

$$q(\mathbf{x}_t|i) = \int q(\mathbf{x}_t|i,\boldsymbol{\theta}_i)q(\boldsymbol{\theta}_i)\,d\boldsymbol{\theta}_i = \int q(\mathbf{x}_t|\mathbf{y}_t,i,\boldsymbol{\theta}_i)q(\mathbf{y}_t|i,\boldsymbol{\theta}_i)q(\boldsymbol{\theta}_i)\,d\mathbf{y}_t\,d\boldsymbol{\theta}_i, \tag{11}$$

with $\boldsymbol{\theta}_i = \{\boldsymbol{\mu}_i,\boldsymbol{\nu}_i,\boldsymbol{\varphi}_i\}$. However, it is difficult to directly compute the above integral over $\{\boldsymbol{\theta}_i,\mathbf{y}_t\}$ for an analytical $q(\mathbf{x}_t|i)$. Therefore, $q(\mathbf{x}_t|i)$ is sequentially approximated by lower-bounds according to Jensen's inequality, i.e.,

$$q(\mathbf{x}_t|i) \ge \int \tilde{q}(\mathbf{x}_t|i,\boldsymbol{\theta}_i)q(\boldsymbol{\theta}_i)\,d\boldsymbol{\theta}_i \ge \int \tilde{q}(\mathbf{x}_t|i,\boldsymbol{\nu}_i,\boldsymbol{\varphi}_i)q(\boldsymbol{\nu}_i)q(\boldsymbol{\varphi}_i)\,d\boldsymbol{\nu}_i\,d\boldsymbol{\varphi}_i, \tag{12}$$

which leads to the marginal $q(\mathbf{x}_t|i)$ as a product of several Student's T-distributions in Eq. (A.1) in Appendix A, where

$$\tilde{q}(\mathbf{x}_t|i,\boldsymbol{\theta}_i) = \exp\left\{\int G(\mathbf{y}_t|\tilde{\mathbf{y}}_{it},\tilde{\mathbf{\Sigma}}_i^{y})\ln\frac{q(\mathbf{x}_t|\mathbf{y}_t,\boldsymbol{\theta}_i)q(\mathbf{y}_t|\boldsymbol{\theta}_i)}{G(\mathbf{y}_t|\tilde{\mathbf{y}}_{it},\tilde{\mathbf{\Sigma}}_i^{y})}d\mathbf{y}\right\}, \tag{13}$$

$$\tilde{q}(\mathbf{x}_t|i,\boldsymbol{\nu}_i,\boldsymbol{\varphi}_i) = \exp\left\{\int G(\boldsymbol{\mu}_i|\tilde{\boldsymbol{\mu}}_{it}^{x},\tilde{\mathbf{\Sigma}}_i^{\mu^x})\ln\frac{\tilde{q}(\mathbf{x}_t|i,\boldsymbol{\theta}_i)q(\boldsymbol{\mu}_i)}{G(\boldsymbol{\mu}_i|\tilde{\boldsymbol{\mu}}_{it}^{x},\tilde{\mathbf{\Sigma}}_i^{\mu^x})}d\boldsymbol{\mu}_i\right\}, \tag{14}$$

and $\{\tilde{\mathbf{y}}_{it},\tilde{\mathbf{\Sigma}}_i^{y},\tilde{\boldsymbol{\mu}}_{it}^{x},\tilde{\mathbf{\Sigma}}_i^{\mu^x}\}$ are assistant variables which can be updated by maximizing the above lower bounds with the details referred to Appendix A.

Similarly for $p(\mathbf{y}|i,\mathbf{x}_t)$, we can also obtain a product of multiple Student's T-distributions, which however makes the subsequent integrals in the harmony measure difficult. We further approximate it by the following Gaussian according to the property of Student's T-distribution [24]:

$$\begin{aligned}
&p(\mathbf{y}|\mathbf{x}_t,i) \approx G(\mathbf{y}|\mathbf{W}_i(\mathbf{x}_t-\tilde{\boldsymbol{\mu}}_{it}^{y}),\varPi_i),\\
&\mathbf{W}_i = \varPi_i\mathbf{U}_i^T\mathbf{D}_i,\quad \mathbf{D}_i = \mathrm{diag}[(\mathbf{a}_i^{\varphi}+\tfrac{1}{2}\mathbf{1}_d)\oslash(\mathbf{b}_i^{\varphi}+\tfrac{1}{2}\ \mathrm{diag}(\tilde{\mathbf{\Sigma}}_i^{\mu^y}))],\\
&\varPi_i = [\mathbf{U}_i^T\mathbf{D}_i\mathbf{U}_i+\mathrm{diag}(\mathbf{a}_i^{\nu}\oslash\mathbf{b}_i^{\nu})]^{-1},
\end{aligned}\tag{15}$$

where $\tilde{\boldsymbol{\mu}}_{it}^{y}$ and $\tilde{\mathbf{\Sigma}}_i^{\mu^y}$ can be updated for a better approximation given the parameters. The details are referred to Appendix A.

Putting Eq. (8) into Eq. (5), the harmony measure on LFA becomes

$$\begin{aligned}
H^{LFA}(\{\mathbf{U}_i\},\Xi) = \int &\sum_{\mathbf{Z}}p(\mathbf{\Theta}|\mathbf{Z},\mathbf{Y},\mathbf{X}_N)p(\mathbf{Y}|\mathbf{Z},\mathbf{X}_N)p(\mathbf{Z}|\mathbf{X}_N)\\
&\cdot\ln[q(\mathbf{X}_N|\mathbf{Y},\mathbf{Z},\mathbf{\Theta})q(\mathbf{Y}|\mathbf{Z},\mathbf{\Theta})q(\mathbf{Z}|\mathbf{\Theta})q(\mathbf{\Theta}|\Xi)]\,d\mathbf{Y}\,d\mathbf{\Theta}.
\end{aligned}\tag{16}$$

By further substituting the details of Eqs. (9)–(15) into Eq. (16), we obtain the following lower-bound:

$$H^{LFA}(\{\mathbf{U}_i\},\Xi) \ge H^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*), \tag{17}$$

where the detailed expression of $H^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*)$ is given by Eq. (A.5) in Appendix A. The best harmony principle is approximated by maximizing $H^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*)$ with respect to the prior hyper-parameters $\Xi = \{\lambda,\xi,\{\mathbf{m}_i\},\beta,\{a_{ij}^{\nu},b_{ij}^{\nu}\},\{a_{ij}^{\varphi},b_{ij}^{\varphi}\}\}$ and the posterior hyper-parameters $\Xi^* = \{\lambda^*,\{\mathbf{m}_i^*\},\{b_{ij}^{*\nu}\},\{b_{ij}^{*\varphi}\}\}$. The derived algorithm is sketched in Table 3 and shortly denoted as BYY-LFA, with details referred to Appendix A.

The above checking on whether to discard dimension j in component i is actually observing whether the jth diagonal element of $\mathbf{\Lambda}_i$ tends to zero and thus the corresponding dimension of $\mathbf{y}$ can be discarded. Following Section 2.2 of [20], e.g., its Eq. (36), this checking can be further improved by the nature of the co-dim matrix pair of each FA model.

Table 3
BYY algorithm on LFA with DNG prior (BYY-LFA).

```
Initialization: Randomly initialize the model with large enough number $k$ of components and hidden dimensionalities $\{h_i\}$; set $\tau=0$ and the harmony measure $J_{BYY}(\tau)=-\infty$;
repeat
    Yang−step: Update assistant variables $\{\tilde{\mathbf{y}}_{it}, \tilde{\mathbf{\Sigma}}_i^y\}$ by Eq. (A.3), $\{\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{\Sigma}}_i^{\mu^x}\}$ by Eq. (A.2), and $\{\tilde{\boldsymbol{\mu}}_{it}^y, \tilde{\mathbf{\Sigma}}_i^{\mu^y}\}$ by Eq. (A.4). Calculate $p(i|\mathbf{x}_t)$ by Eq. (A.1), $\mathbf{e}_{it}$ and $\epsilon_{it}$ by Eq. (A.6).
    Ying−step: With $\nabla_{\mathbf{U}_i} H^{LFA}$ by Eq. (A.8), update each $\mathbf{U}_i$ via
        $\mathbf{U}_i^{new} = \mathbf{U}_i^{old} + \eta[\nabla_{\mathbf{U}_i} H^{LFA} - \mathbf{U}_i^{old}(\nabla_{\mathbf{U}_i} H^{LFA})^T \mathbf{U}_i^{old}]$.
      Update $\mathbf{\Xi}^* = \{\lambda_i^*, \mathbf{m}_i^*, \mathbf{b}_i^{*\nu}, \mathbf{b}_i^{*\varphi}\}$ in a gradient way by Eq. (A.7).
    H−step: In a gradient way, update $\{\lambda, \xi, \beta\} \cup \{\mathbf{m}_i\}$ by Eq. (A.10), $\{\mathbf{a}_i^\nu, \mathbf{b}_i^\nu\}$ by Eq. (A.11), and $\{\mathbf{a}_i^\varphi, \mathbf{b}_i^\varphi\}$ by Eq. (A.12).
    $\otimes$ for $i=1,\dots,k$ do
        if $\lambda_i^*$ or $\lambda_i \to 0$ then discard component $i$, let $k=k-1$;
        for $j=1,\dots,h_i$ do
            if $\frac{b_{ij}^{*\nu}}{a_{ij}^\nu + \frac{1}{2}\sum_{t=1}^N p(i|\mathbf{x}_t)}$ or $\frac{b_{ij}^\nu}{a_{ij}^\nu} \to 0$ then discard dimension $j$ in component $i$, let $h_i = h_i - 1$;
    if another 5 runs pass then let $\tau=\tau+1$; calculate $J_{BYY}(\tau) = H^{LFA}$ by Eq. (A.5);
until $J_{BYY}(\tau) - J_{BYY}(\tau-1) < \epsilon J_{BYY}(\tau-1)$, with $\epsilon = 10^{-5}$;
```

3.3. BYY algorithm for learning MFA with DNG prior

Learning on MFA can be made in a similar way. The main differences come from the parameterizations on the factor loading matrix and the factor covariance matrix, and their corresponding prior distributions, as shown in Tables 1 and 2.

The Ying machine is represented by replacing the counterparts of LFA with the ones of MFA, i.e., getting $q(\mathbf{X},\mathbf{Y},\mathbf{Z}|\mathbf{\Theta})$ in Eq. (3) by the left column of Table 1 and $q(\mathbf{\Theta})$ by the left part of Table 2. In Yang machine, we still consider the empirical distribution $p(\mathbf{X}) = \delta(\mathbf{X}-\mathbf{X}_N)$, but the Yang-pathway $p(\mathbf{R}|\mathbf{X})$ is factorized in a form slightly different from Eq. (8):

$$p(\mathbf{R}|\mathbf{X}_N) = p(\mathbf{\Theta}|\mathbf{Z},\mathbf{X}_N)p(\mathbf{Y}|\mathbf{Z},\mathbf{X}_N,\{\mathbf{A}_i\})p(\mathbf{Z}|\mathbf{X}_N),$$

$$p(\mathbf{\Theta}|\mathbf{Z},\mathbf{X}_N) = p(\boldsymbol{\alpha}|\mathbf{Z},\mathbf{X}_N)\prod_{i=1}^{k}\left\{p(\boldsymbol{\mu}_i|\mathbf{Z},\mathbf{X}_N)\cdot\prod_{j=1}^{d}p(\varphi_i^{(j)}|\mathbf{Z},\mathbf{X}_N)\right.$$
$$\left.\cdot\prod_{j=1}^{h_i}[p(\mathbf{A}_i^{(*j)}|\mathbf{Z},\mathbf{X}_N)p(\varsigma_i^{(j)}|\mathbf{Z},\mathbf{X}_N)]\right\},$$

$$p(\mathbf{Y}|\mathbf{Z},\mathbf{X}_N,\{\mathbf{A}_i\}) = \prod_{i=1}^{k}\prod_{t=1}^{N}p(\mathbf{y}|i,\mathbf{x}_t,\mathbf{A}_i)^{z_{it}}, \tag{18}$$

where $p(\mathbf{Z}|\mathbf{X}_N)$ is expressed in the same form as the one in Eq. (8). We proceed to the detailed expressions of $p(\mathbf{\Theta}|\mathbf{Z},\mathbf{X}_N)$ according to the conjugate property of the DNG priors on MFA. Particularly, Eq. (8) is modified accordingly by replacing $p(\nu_i^{(j)}|\mathbf{Y},\mathbf{Z},\mathbf{X}_N)$ with the following equations:

$$p(\mathbf{A}_i^{(*j)}|\mathbf{Z},\mathbf{X}_N) = G\left(\mathbf{A}_i^{(*j)}|\overline{\mathbf{A}}_i^{(*j)}, \left[a_{ij}^\varsigma/b_{ij}^\varsigma\mathbf{I}_d + \left(\sum_{t=1}^{N}z_{it}\right)\mathrm{diag}(\mathbf{a}_i^\varphi \oslash \mathbf{b}_i^\varphi)\right]^{-1}\right),$$

$$p(\varsigma_i^{(j)}|\mathbf{Z},\mathbf{X}_N) = \Gamma\left(\varsigma_i^{(j)}|a_{ij}^\varsigma + \frac{d}{2}, b_{ij}^{*\varsigma}\right), \tag{19}$$

where $\{\overline{\mathbf{A}}_i^{(*j)}, b_{ij}^{*\varsigma}, b_{ij}^{*\varphi}\}$ are free hyper-parameters to be determined.

Similar to Eq. (10), the $p(i|\mathbf{x}_t)$ is constructed by the Bayesian posterior, i.e., $p(i|\mathbf{x}_t) \propto q(\mathbf{x}_t,i)$. The $q(\mathbf{x}_t|i)$ for MFA can also be approximated via Eqs. (12)–(14), where $\tilde{q}(\mathbf{x}_t|i,\boldsymbol{\nu}_i,\boldsymbol{\varphi}_i)$ is substituted by $\tilde{q}(\mathbf{x}_t|i,\mathbf{A}_i,\boldsymbol{\varphi}_i)$, and then Eq. (12) is computed by integrating $\tilde{q}(\mathbf{x}_t|i,\mathbf{A}_i,\boldsymbol{\varphi}_i)q(\zeta_i)q(\boldsymbol{\varphi}_i)$ with respect to $\mathbf{A}_i,\zeta_i,\boldsymbol{\varphi}_i$, leading to a different $q(\mathbf{x}_t|i)$ which is also a product of multiple Students T-distributions. Moreover, the $p(\mathbf{y}|i,\mathbf{x}_t,\{\mathbf{A}_i\})$ is approximated by Eq. (15) with slight modifications: replacing $\mathbf{U}_i$ and $\mathrm{diag}(\mathbf{a}_i^\nu \oslash \mathbf{b}_i^\nu)$ with $\mathbf{A}_i$ and $\mathbf{I}_{h_i}$ respectively.

Putting the above specifications of Ying-Yang machine into Eq. (5), we obtain a lower-bound $H^{MFA}(\mathbf{\Xi},\mathbf{\Xi}^*)$ analogous to $H^{LFA}(\{\mathbf{U}_i\},\mathbf{\Xi},\mathbf{\Xi}^*)$, and also a corresponding BYY-MFA algorithm to maximize $H^{MFA}(\mathbf{\Xi},\mathbf{\Xi}^*)$ via modifying the counterparts of BYY-LFA according to Eqs. (18) and (19). Readers are referred to [25] for details.

It should be noted that the $p(\boldsymbol{\theta}_i|\mathbf{Y},\mathbf{Z},\mathbf{X}_N)$ is considered to have a conjugate Normal–Gamma form, whereas it was approximated by a 2nd order Taylor expansion in [9] (see Eqs. (13)–(17) and Table B1 in [9] for details), although the prior $q(\boldsymbol{\theta}_i)$ on each component's parameters $\boldsymbol{\theta}_i$ in Table 2 is the same as those in [9]. Therefore, when the mixture model in Eq. (1) degenerates to only one Gaussian component represented by FA in [9], i.e., $k=1$, the BYY-LFA and BYY-MFA here actually provide alternative BYY learning algorithms for FA-a and FA-b, being different from the algorithms derived in [9].

4. VB algorithms for learning LFA versus MFA

With proper incorporation of prior knowledge on model parameters, Bayesian model selection is implemented via the maximum marginal likelihood, which is obtained by integrating out the latent variables $\mathbf{Y}$ and the parameters $\mathbf{\Theta}$, i.e., $q(\mathbf{X}_N) = \int q(\mathbf{X}_N,\mathbf{Y}|\mathbf{\Theta})q(\mathbf{\Theta})\,d\mathbf{Y}\,d\mathbf{\Theta}$. However, the involved integration is usually very difficult. Variational Bayesian [6,21] tackles this difficulty via constructing a tractable lower bound for the log marginal likelihood by means of variational methods, and an EM-like algorithm is employed to optimize this lower bound. More precisely, the lower bound is given as follows:

$$J_{VB}(\mathbf{\Xi}^*,\mathbf{\Xi}) = \int p(\mathbf{\Theta},\mathbf{Y}|\mathbf{\Xi}^*)\ln\frac{q(\mathbf{X}_N,\mathbf{Y}|\mathbf{\Theta})q(\mathbf{\Theta}|\mathbf{\Xi})}{p(\mathbf{\Theta},\mathbf{Y}|\mathbf{\Xi}^*)}\,d\mathbf{Y}\,d\mathbf{\Theta}$$
$$= \ln q(\mathbf{X}_N|\mathbf{\Xi}) - \int p(\mathbf{\Theta},\mathbf{Y}|\mathbf{\Xi}^*)\ln\frac{p(\mathbf{\Theta},\mathbf{Y}|\mathbf{\Xi}^*)}{p(\mathbf{\Theta},\mathbf{Y}|\mathbf{X}_N,\mathbf{\Xi})}\,d\mathbf{Y}\,d\mathbf{\Theta}, \tag{20}$$

where $\mathbf{Y}$ represents all hidden variables, e.g., $\{\mathbf{Y},\mathbf{Z}\}$ in Eq. (3) for MFA/LFA, $q(\mathbf{\Theta}|\mathbf{\Xi})$ is a prior on parameters $\mathbf{\Theta}$ with hyper-parameters $\mathbf{\Xi}$, and $p(\mathbf{\Theta},\mathbf{Y}|\mathbf{\Xi}^*)$ is a variational posterior with hyper-parameters $\mathbf{\Xi}^*$ to approximate the exact Bayesian posterior $p(\mathbf{\Theta},\mathbf{Y}|\mathbf{X}_N,\mathbf{\Xi}) \propto q(\mathbf{X}_N,\mathbf{Y}|\mathbf{\Theta})q(\mathbf{\Theta}|\mathbf{\Xi})$. It follows from Eq. (20) that the lower bound $J_{VB}(\mathbf{\Xi}^*,\mathbf{\Xi})$ is tight to $\ln q(\mathbf{X}_N|\mathbf{\Xi})$ when $p(\mathbf{\Theta},\mathbf{Y}|\mathbf{\Xi}^*) = p(\mathbf{\Theta},\mathbf{Y}|\mathbf{X}_N,\mathbf{\Xi})$. For computational convenience, $q(\mathbf{\Theta}|\mathbf{\Xi})$ is usually chosen to be conjugate priors, and $p(\mathbf{\Theta},\mathbf{Y})$ is usually assumed to be afactorized form $p(\mathbf{\Theta},\mathbf{Y}) = p(\mathbf{Y})\prod_i p(\mathbf{\Theta}_i)$ with $\mathbf{\Theta} = \bigcup_i \mathbf{\Theta}_i$, so that maximizing J_{VB} makes the variational posterior $p(\mathbf{\Theta})$ to be conjugate, i.e., in the same form as the corresponding prior.

There is already a VB learning algorithm on MFA proposed in [6], where the prior on $\boldsymbol{\theta}_i$ is $q(\boldsymbol{\theta}_i) = q(\boldsymbol{\varphi}_i)q(\mathbf{A}_i)$, without considering the prior on $\boldsymbol{\mu}_i$. In this paper, the DNG prior in Table 2 considers a Gaussian distribution $q(\boldsymbol{\mu}_i)$ for $\boldsymbol{\mu}_i$. Then, the corresponding VB algorithm (shortly denoted as VB-MFA) can be obtained through slightly modifying the one in [6] by replacing $\boldsymbol{\mu}_i$ with $\mathbf{m}_i^*$ when updating other parameters, and additionally update the following variational posterior for $\boldsymbol{\mu}_i$:

$$p(\boldsymbol{\mu}_i) = G(\boldsymbol{\mu}_i|\mathbf{m}_i^*, \boldsymbol{\Sigma}_i^{*\mu}) \propto E_p[\ln[q(\mathbf{X}_N, \mathbf{Y}, \mathbf{Z}|\boldsymbol{\Theta})q(\boldsymbol{\Theta})]], \tag{21}$$

where $E_p[\cdot]$ denotes expectation with respect to the current estimate of all variational posteriors $p(\mathbf{Y}|\mathbf{Z})p(\mathbf{Z})p(\boldsymbol{\Theta})$ except $p(\boldsymbol{\mu}_i)$.

There is no VB algorithm available for LFA with DNG prior, yet there is a VB algorithm presented in [9] on FA-b, a degenerated case of LFA with $k=1$. Following [6], the joint posterior is assumed to be approximately factorized as the following variational posterior:

$$\begin{aligned} &p(\mathbf{Y}, \mathbf{Z}, \boldsymbol{\Theta}|\mathbf{X}_N) \approx p(\mathbf{Y}|\mathbf{Z})p(\mathbf{Z})p(\boldsymbol{\Theta}), \\ &\quad p(\boldsymbol{\Theta}) = p(\boldsymbol{\alpha}) \prod_{i=1}^{k} \left[p(\boldsymbol{\mu}_i) \prod_{j=1}^{h_i} p(\nu_i^{(j)}) \prod_{j=1}^{d} p(\varphi_i^{(j)}) \right], \end{aligned} \tag{22}$$

and then the variational lower bound becomes

$$J_{VB}^{LFA} = \int p(\mathbf{Y}|\mathbf{Z})p(\mathbf{Z})p(\boldsymbol{\Theta}) \ln\left[\frac{q(\mathbf{X}, \mathbf{Y}, \mathbf{Z}|\boldsymbol{\Theta})q(\boldsymbol{\Theta})}{p(\mathbf{Y}|\mathbf{Z})p(\mathbf{Z})p(\boldsymbol{\Theta})}\right] d\mathbf{Y}\, d\mathbf{Z}\, d\mathbf{X}\, d\boldsymbol{\Theta}. \tag{23}$$

Table 4
Four series of experiments for MFA/LFA model selection.

Starting case	$(N, d, k^*, \beta) = (300, 10, 3, 0.1)$
Series 1	Vary $N \in \{300, 290, 280, \ldots, 100\}$ and fix d, k^*, β
Series 2	Vary $d \in \{10, 12, 14, \ldots, 30\}$ and fix N, k^*, β
Series 3	Vary $k^* \in \{3, 4, 5, \ldots, 15\}$ and fix N, d, β
Series 4	Vary $\beta \in \{0.1, 0.2, \ldots, 1.5\}$ and fix N, d, k^*

Its maximization leads to the following conjugate form:

$$\begin{aligned} &p(\mathbf{Z}) = \prod_{i=1}^{k} \prod_{t=1}^{N} p_{it}^{z_{it}}, \quad p(\mathbf{Y}|\mathbf{Z}) = \prod_{i=1}^{k} \prod_{t=1}^{N} G(\mathbf{y}|\overline{\mathbf{y}}_{it}^*, \boldsymbol{\Sigma}_i^{*y})^{z_{it}}, \\ &\quad p(\boldsymbol{\alpha}) = \mathcal{D}(\boldsymbol{\alpha}|\lambda^*, \xi^*), \quad p(\boldsymbol{\mu}_i) = G(\boldsymbol{\mu}_i|\mathbf{m}_i^*, \boldsymbol{\Sigma}_i^{*\mu}), \\ &\quad p(\nu_i^{(j)}) = \Gamma(\nu_i^{(j)}|a_{ij}^{*\nu}, b_{ij}^{*\nu}), \quad p(\varphi_i^{(j)}) = \Gamma(\varphi_i^{(j)}|a_{ij}^{*\varphi}, b_{ij}^{*\varphi}). \end{aligned} \tag{24}$$

An EM-like algorithm is given in Appendix B, for maximizing J_{VB}^{LFA} with respect to the variational parameters $\{\{p_{it}, \overline{\mathbf{y}}_{it}^*\}_t, \boldsymbol{\Sigma}_i^{*y}\}_{i=1}^k$ that describe the posterior distribution on $\mathbf{Y} \cup \mathbf{Z}$, and the variational hyper-parameters $\boldsymbol{\Xi}^* = \{\lambda^*, \xi^*, \{\mathbf{m}_i^*, \boldsymbol{\Sigma}_i^{*\mu}\}_i, \{a_{ij}^{*\nu}, b_{ij}^{*\nu}\}_{i,j}, \{a_{ij}^{*\varphi}, b_{ij}^{*\varphi}\}_{i,j}\}$ that describe the posterior on $\boldsymbol{\Theta}$.

5. Experimental results

Since the values of the hyper-parameters $\boldsymbol{\Xi}$ are usually unknown and it is not good to assign one for all data by guess, the algorithms are implemented to adjust $\boldsymbol{\Xi}$ under their own learning principles. Moreover, it has been demonstrated in [4] that the performance of VB will be improved when the $\{\mathbf{m}_i\}$ of Table 2 are constrained to be the same, i.e., $\forall i, \mathbf{m}_i = \mathbf{m}$, with the number of free hyper-parameters reduced, and thus VB is here implemented with $\mathbf{m}_i = \mathbf{m}$ too. Still, no constraints are imposed on $\{\mathbf{m}_i\}$ for BYY algorithms.

5.1. Comparisons on four series of simulations

Each dataset is generated according to MFA or LFA in Tables 1 and 2. For each component i, we set the hidden dimensionality $h_i^* = 5$, and the mixing weight $\alpha_i = 1/k^*$, where k^* denotes the true component number. The remaining parameters are randomly generated according to the Normal–Gamma distributions given in Table 2, with $a_{ij}^{\varsigma} = b_{ij}^{\varsigma} = 3$, $a_{ij}^{\nu} = 10$, $b_{ij}^{\nu} = 200$, and $a_{ij}^{\varphi} = b_{ij}^{\varphi} = 10$, $\forall i, j$.

To cover a wide range of experimental conditions, we vary the values of the sample size N, the observed data dimensionality d, the true component number k^*, and the overlap degree β of Gaussian components, where increasing β indicates that the degree of separation of the components changes from large to small. Generally speaking, it becomes more difficult in model selection as N decreases, d increases, k^* increases, and β increases.

We consider four series of experiments specified in Table 4. Starting from a same point in the 4-dimensional factor space, each series varies one factor of (N, d, k^*, β) while fixing the remaining three. For each specific setting, 500 datasets are generated independently, and all algorithms are initialized with a same component number 25 and a same hidden dimensionality 9. We compare the resulted $k \cup \{h_i\}_{i=1}^k$ with the true $k^* \cup \{h_i^*\}_{i=1}^{k^*}$ with each $h_i^* = 5$. The model selection accuracies are reported in Fig. 1 as percentages of correctly obtaining $k = k^*$ and $h_i = 5(\forall i)$ out of 500 independent runs.

Fig. 1 shows that all algorithms perform well at the starting case, and then decline as the experimental environment deteriorates. Particularly, we observe:

1. LFA is shown to be superior to MFA in model selection, under either BYY or VB. This observation is consistent to the empirical findings on FA in [9]. Specifically, the superiority of LFA is less obvious in the cases of small N, or large k^*, β, because both LFA and MFA get bad performance on these extreme cases. The reason may be due to the fact that $\mathbf{y}$ in MFA processes an identity covariance matrix, which can be taken as a special case of that of LFA. Thus, LFA is more flexible than MFA to accommodate different types of data. Moreover, the LFA is better than MFA in terms of providing one additional room for model selection via estimating Λ. Compared with adding priors, $G(\mathbf{y}|0, \Lambda)$ is more reliable and easier to be estimated from data.
2. On either LFA or MFA, BYY greatly outperforms VB for all the cases except the one $k^* = 7$ in series 3. Although VB-LFA benefits from the superiority of using LFA instead of MFA, BYY-MFA is still more robust than VB-LFA against the deterioration of the environment, while BYY-LFA is the best in general.

Regarding the time-cost, due to space limit we are not able to present the detailed results. It was observed that the BYY in general involves a heavier computational load than VB because it involves more gradient calculations. However, the complexities of both algorithms are comparable to each other (around $\mathcal{O}(N^3)$), since the main computation of both of them arises from the matrix multiplication.

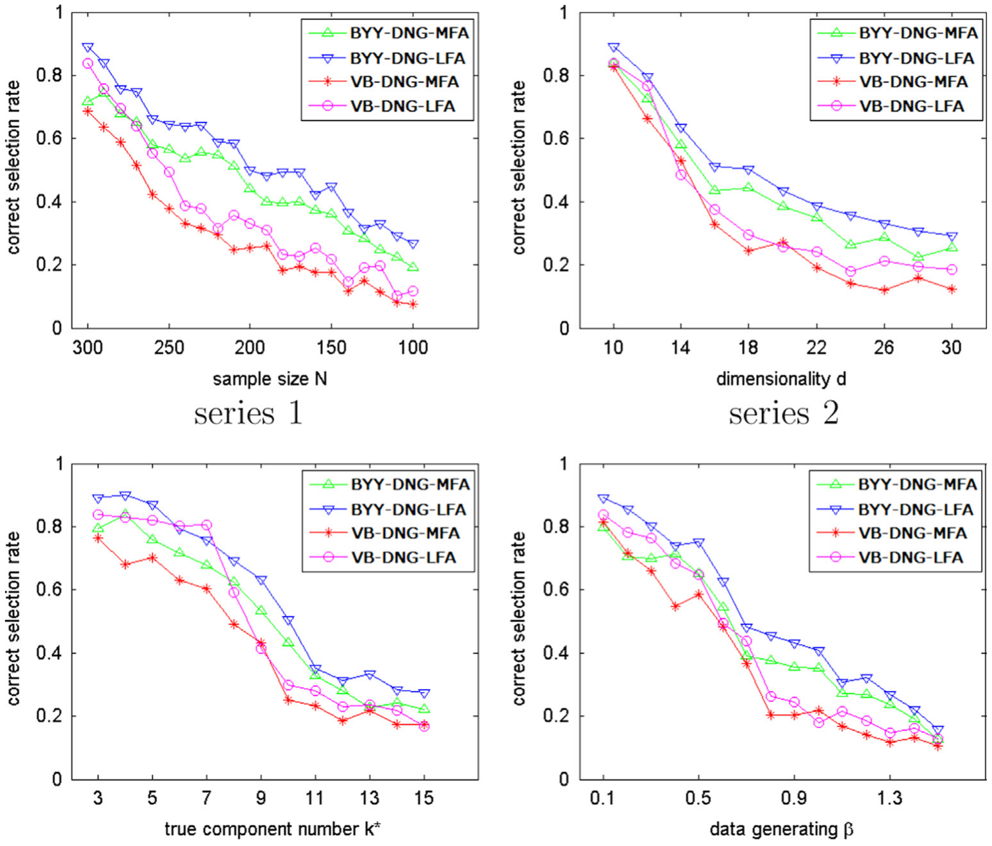
5.2. Face and handwritten digit images clustering

We test the clustering performances of the four algorithms on three real datasets: the ORL[1] face image database, the USPS[2] handwritten digits, and the MNIST[3] handwritten digits. The ORL contains with 10 grayscale images for each 40 human subjects, and all images are of a size 64 × 64. We project them to 65 dimensions by PCA so that 90% energy in the covariance is reserved. The USPS and MNIST digit databases both contain grayscale images of "0" through "9". In USPS, each image is of a size 16 × 16 and thus 256-dimensional, and there are 1100 images for each digit. In MNIST, each image is of a size 28 × 28 and thus 784-dimensional, and

[1] Downloaded from "http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html".

[2] Downloaded from "http://cs.nyu.edu/~roweis/data.html".

[3] Downloaded from "yann.lecun.com/exdb/mnist/".



series 1

series 2

series 3

series 4

Fig. 1. The correct selection rates by MFA/LFA automatic model selection algorithms in the four series of simulations.

each digit has 12,000 images. To study model selection on a small sample size, we randomly pick 1200 images for each digit in MNIST.

After MFA/LFA learning in an unsupervised way, all samples are partitioned into different clusters with each component being a cluster. Since we do not know the true component number in these real world databases, we use two metrics to evaluate the clustering performance: (1) Rand index (RI) [26]; (2) normalized mutual information (NMI) [27]. Both metrics compare the clustered partition with a ground truth partition, which is formed by letting each class (i.e., each person in ORL and each digit in USPS/MNIST) be a cluster. Both RI and NMI take value in [0, 1] and equal to 1 when two partitions are identical, and a higher value means greater similarity between the obtained clusters and the ground truth [28]. It should be also noted that, since an appropriate model selection helps improve the generalization ability, a high RI/NMI score is related to but does not necessarily come from an appropriately determined component number.

After 10 independent runs for each algorithm, the average RI and NMI scores by using each algorithm are reported in Table 5. As shown, on both MFA and LFA, BYY outperforms VB in terms of both RI and NMI scores. Moreover, LFA provides better results than MFA under BYY or VB. Out of all algorithms, BYY-LFA performs the best.

Table 5
Clustering performance by MFA/LFA algorithms on ORL, USPS and MNIST databases.

Index per algorithm	ORL	USPS	MNIST
Rand index (RI)			
BYY-MFA	0.90	0.87	0.87
BYY-LFA	0.92	0.91	0.90
VB-MFA	0.86	0.84	0.86
VB-LFA	0.89	0.88	0.89
Normalized mutual information (NMI)			
BYY-MFA	0.91	0.88	0.89
BYY-LFA	0.93	0.91	0.91
VB-MFA	0.89	0.87	0.85
VB-LFA	0.91	0.88	0.88

5.3. Unsupervised image segmentation

We also apply the algorithms to unsupervised image segmentation on the Berkeley segmentation database of real world images[4]. Based on the fact that any feature of a single pixel may not be sufficient for segmentation, we choose the VZ features proposed in [29,30] to take into account the neighborhood and texture information. A VZ feature for each pixel is constructed by vectorizing the color information of all pixels in a $w \times w$-sized window centered at the pixel. Here, we set $w=7$ to construct 147-dimensional feature vectors from the LAB color space. Usually, the VZ features are further projected to 8 dimensions by PCA, e.g., in [4,29,30], when the dimensionality is too high. Since MFA/LFA can simultaneously perform clustering and local dimensionality reduction, we use the 147 features directly without PCA preprocessing as PCA may result in some information loss.

For every image, the trained MFA/LFA model assigns each image pixel to the cluster (represented by a component) of maximum posterior probability. Since the true component number is unknown, we evaluate the resulted segmentations by the Probabilistic Rand (PR) index [31], which takes values between 0 and 1. A higher PR score indicates a better segmentation with a

[4] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/.

Table 6
Average PR scores of 5 runs on the 100 testing images of Berkeley image segmentation database by MFA/LFA algorithms on the 147-dimensional features. The " GMM" results are obtained by GMM automatic model selection algorithms on the 8-dimensional VZ features in [4].

GMM from [4]		MFA/LFA			
VB-DNW	BYY-DNW	VB-MFA	VB-LFA	BYY-MFA	BYY-LFA
0.803	0.851	0.819	0.845	0.864	0.878

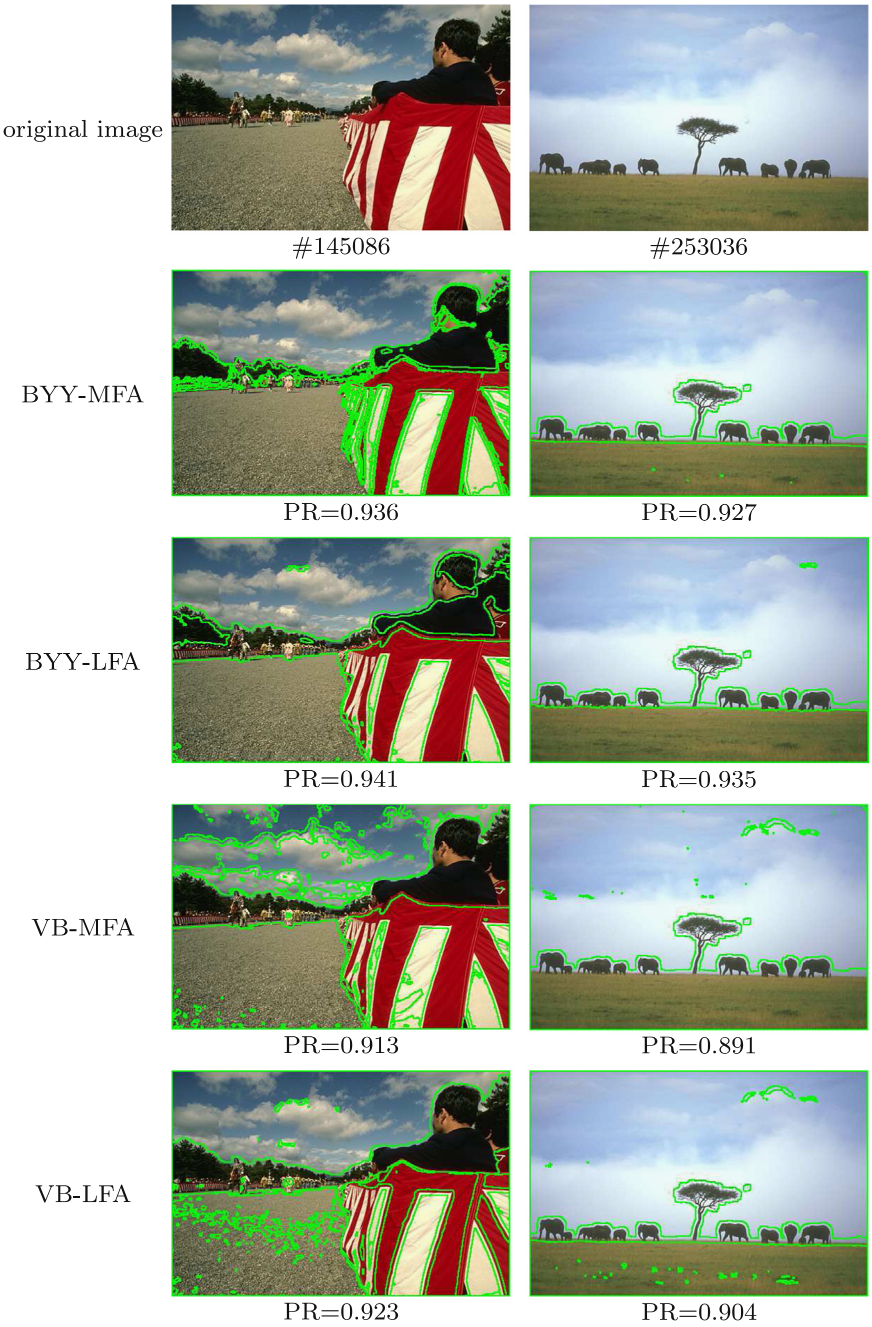


Fig. 2. Two image segmentation examples from Berkeley segmentation database by MFA/LFA automatic model selection algorithms. The segments are illustrated by the highlighted green boundaries. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

higher percentage of pixel pairs in the segmentation having the same labels as in the ground truth segmentation. A good model selection performance is closely related to, but not necessarily implies, a high PR score. We directly compare the segmentation performance of the four algorithms without any post-processings such as region merging and graph cut [32], although these techniques may further improve the segmentations.

Table 6 gives the average PR scores by 5 runs on all of the 100 testing images of the Berkeley image segmentation database. Shown in Fig. 2 are two examples chosen from the database. On both MFA and LFA, the BYY algorithm shows a higher average PR score than VB, which is consistent with their model selection performances on simulated data. Also, the BYY-LFA algorithm performs the best and is able to detect the objects of interest from a confusing background.

Moreover, compared with the results in [4] by GMM algorithms on the PCA preprocessed 8-dimensional VZ samples, learning MFA/LFA based on the original 147-dimensional samples provides further improvements for both BYY and VB. This may show that it is advantageous to jointly perform clustering and local subspace modeling in image segmentation.

6. Concluding remarks

This paper has presented a comparative investigation on the relative strengths and weaknesses of VB and BYY in automatic model selection on MFA and LFA with the conjugate DNG priors. The algorithm in [6] for VB on MFA is slightly modified. Moreover, not only the algorithm for VB on LFA is developed, but also the algorithms for BYY on MFA and LFA are proposed.

Through synthetic experiments, we have the following empirical findings. First, LFA performs better than MFA for both VB and BYY, which echoes the advantages of FA-b over FA-a observed in [9]. Second, BYY outperforms VB on both MFA and LFA. Overall, the BYY-LFA algorithm performs the best in most cases. These observations are reconfirmed by applications on not only face and handwritten digit images clustering, but also unsupervised image segmentation on real world images.

Acknowledgments

The authors thank the anonymous reviewers whose comments greatly improved the manuscript. This work was fully supported by RGC Direct (grant 4055025), and partially supported by the National Science Foundation of China (NSFC) (grant 61375005), and the National Basic Research Program of China (973 Program) (grant 2009CB825404).

Appendix A. The BYY-LFA algorithm

The $q(\mathbf{x}_t|i)$ is approximated by Eq. (12) with the following product of multiple Student's T-distributions:

$$q(\mathbf{x}_t|i) \approx \frac{\beta^{d/2}\exp\left(\frac{d+h_i}{2}\right)|\tilde{\boldsymbol{\Sigma}}_i^{\mu^x}|^{1/2}|\tilde{\boldsymbol{\Sigma}}_i^{y}|^{1/2}}{(2\pi)^{d/2}\exp\left\{\frac{\beta}{2}\mathrm{tr}[\tilde{\boldsymbol{\Sigma}}_i^{\mu^x}+(\tilde{\boldsymbol{\mu}}_{it}^x-\mathbf{m}_i)(\tilde{\boldsymbol{\mu}}_{it}^x-\mathbf{m}_i)^T]\right\}} \cdot \prod_{j=1}^{h_i}\frac{(b_{ij}^{\nu})^{a_{ij}^{\nu}}\cdot\Gamma\left(a_{ij}^{\nu}+\frac{1}{2}\right)}{\Gamma(a_{ij}^{\nu})\left[b_{ij}^{\nu}+\frac{1}{2}\ \mathrm{diag}(\tilde{\boldsymbol{\Sigma}}_i^{y}+\tilde{\mathbf{y}}_{it}\tilde{\mathbf{y}}_{it}^T)^{(j)}\right]^{(a_{ij}^{\nu}+1/2)}} \cdot \prod_{j=1}^{d}\frac{(b_{ij}^{\varphi})^{a_{ij}^{\varphi}}\cdot\Gamma(a_{ij}^{\varphi}+\frac{1}{2})}{\Gamma(a_{ij}^{\varphi})\left[b_{ij}^{\varphi}+\frac{1}{2}\sigma_{itj}\right]^{(a_{ij}^{\varphi}+1/2)}}, \tag{A.1}$$

where $\sigma_{itj}=\mathrm{diag}(\mathbf{Q}_{it})^{(j)}$, $\mathbf{Q}_{it}=(\mathbf{x}_t-\mathbf{U}_i\tilde{\mathbf{y}}_{it}-\tilde{\boldsymbol{\mu}}_{it}^x)(\mathbf{x}_t-\mathbf{U}_i\tilde{\mathbf{y}}_{it}-\tilde{\boldsymbol{\mu}}_{it}^x)^T+\tilde{\boldsymbol{\Sigma}}_i^{\mu^x}+\mathbf{U}_i\tilde{\boldsymbol{\Sigma}}_i^{y}\mathbf{U}_i^T$, and the notations $\tilde{\mathbf{y}}_{it}$, $\tilde{\boldsymbol{\Sigma}}_i^{y}$, $\tilde{\boldsymbol{\mu}}_{it}^x$, $\tilde{\boldsymbol{\Sigma}}_i^{\mu^x}$ are assistant variables for approximations and updated by

$$\tilde{\boldsymbol{\mu}}_{it}^x=\tilde{\boldsymbol{\Sigma}}_i^{\mu^x}[\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})(\mathbf{x}_t-\mathbf{U}_i\tilde{\mathbf{y}}_{it})+\beta\mathbf{m}_i],\quad \tilde{\boldsymbol{\Sigma}}_i^{\mu^x}=[\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})+\beta\mathbf{I}_d]^{-1}, \tag{A.2}$$

$$\tilde{\mathbf{y}}_{it}=\tilde{\boldsymbol{\Sigma}}_i^{y}\mathbf{U}_i^T\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})(\mathbf{x}_t-\mathbf{m}_i),\quad \tilde{\boldsymbol{\Sigma}}_i^{y}=[\mathbf{U}_i^T\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})\mathbf{U}_i+\mathrm{diag}(\mathbf{a}_i^{\nu}\oslash\mathbf{b}_i^{\nu})]^{-1}, \tag{A.3}$$

$$\tilde{\boldsymbol{\mu}}_{it}^y=\tilde{\boldsymbol{\Sigma}}_i^{\mu^y}[\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})\mathbf{x}_t+\beta\mathbf{m}_i],\quad \tilde{\boldsymbol{\Sigma}}_i^{\mu^y}=[\mathrm{diag}(\mathbf{a}_i^{\varphi}\oslash\mathbf{b}_i^{\varphi})+\beta\mathbf{I}_d]^{-1}. \tag{A.4}$$

The $H^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*)$ in Eq. (17) is expressed as

$$\begin{aligned} H^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*) &= -\frac{Nd}{2}\ln(2\pi)+\frac{1}{2}\sum_{i=1}^{k}\sum_{t=1}^{N}p(i|\mathbf{x}_t)L_{it}^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*) \\ &\quad +\sum_{i=1}^{k}R_i^{LFA}(\Xi,\Xi^*), \\ L_{it}^{LFA}(\{\mathbf{U}_i\},\Xi,\Xi^*) &= -h_i\ln(2\pi)+\Psi(\lambda_i^*(\xi+N))-\Psi(\xi+N) \\ &\quad -\mathrm{tr}[\mathrm{diag}(\boldsymbol{\omega}_{it}^{*\nu})(\mathbf{W}_i\mathbf{e}_{it}\mathbf{e}_{it}^T\mathbf{W}_i^T+\Pi_i)] \\ &\quad -\mathrm{tr}[\mathrm{diag}(\boldsymbol{\omega}_{it}^{*\varphi})(\boldsymbol{\epsilon}_{it}\boldsymbol{\epsilon}_{it}^T+\mathbf{U}_i\Pi_i\mathbf{U}_i^T+\mathbf{I}_d/\beta)], \end{aligned}$$

$$\begin{aligned} R_i^{LFA}(\Xi,\Xi^*) &= \frac{1}{k}\ln\Gamma(\xi)-\ln\Gamma(\xi\lambda_i)+(\xi\lambda_i-1)[\Psi(\lambda_i^*(\xi+N))-\Psi(\xi+N)] \\ &\quad +\frac{1}{2}[-d\ln(2\pi)+d\ln\beta-\beta(\mathbf{m}_i^*-\mathbf{m}_i)^T(\mathbf{m}_i^*-\mathbf{m}_i)-d] \\ &\quad +\mathbf{a}_i^{\nu}T\ln\mathbf{b}_i^{\nu}-\mathbf{1}_{h_i}^T\ln\Gamma(\mathbf{a}_i^{\nu})+\left(\mathbf{a}_i^{*\nu}-\frac{1}{2}\mathbf{1}_{h_i}\right)^T(\Psi(\mathbf{a}_i^{\nu}) \\ &\quad -\ln\mathbf{b}_i^{*\nu})-\mathbf{b}_i^{\nu}T(\mathbf{a}_i^{*\nu}\oslash\mathbf{b}_i^{*\nu})+\mathbf{a}_i^{\varphi}T\ln\mathbf{b}_i^{\varphi}-\mathbf{1}_{h_i}^T\ln\Gamma(\mathbf{a}_i^{\varphi}) \\ &\quad +\left(\mathbf{a}_i^{*\varphi}-\frac{1}{2}\mathbf{1}_d\right)^T(\Psi(\mathbf{a}_i^{\varphi})-\ln\mathbf{b}_i^{*\varphi})-\mathbf{b}_i^{\varphi}T(\mathbf{a}_i^{*\varphi}\oslash\mathbf{b}_i^{*\varphi}), \end{aligned} \tag{A.5}$$

with the prior hyper-parameters $\Xi=\{\lambda,\xi,\{\mathbf{m}_i\},\beta,\{a_{ij}^{\nu},b_{ij}^{\nu}\},\{a_{ij}^{\varphi},b_{ij}^{\varphi}\}\}$ and the posterior hyper-parameters $\Xi^*=\{\lambda^*,\{\mathbf{m}_i^*\},\{b_{ij}^{*\nu}\},\{b_{ij}^{*\varphi}\}\}$, and we adopt the following denotations for convenience. Table 3 gives an algorithm to maximize the above lower bound in gradient, i.e., BYY-LFA.

$$\begin{aligned} &\mathbf{a}_i^{*\nu}=\mathbf{a}_i^{\nu}+\frac{1}{2}\sum_{t=1}^{N}p(i|\mathbf{x}_t)\mathbf{1}_{h_i},\quad \mathbf{a}_i^{*\varphi}=\mathbf{a}_i^{\varphi}+\frac{1}{2}\sum_{t=1}^{N}p(i|\mathbf{x}_t)\mathbf{1}_d, \\ &\mathbf{e}_{it}=\mathbf{x}_t-\tilde{\boldsymbol{\mu}}_{it}^y,\quad \boldsymbol{\epsilon}_{it}=\mathbf{x}_t-\mathbf{U}_i\mathbf{W}_i\mathbf{e}_{it}-\mathbf{m}_i^*, \\ &\boldsymbol{\omega}_{it}^{*\nu}=[\mathbf{a}_i^{*\nu}+(1-p(i|\mathbf{x}_t))\mathbf{1}_{h_i}/2]\oslash\mathbf{b}_i^{*\nu}, \\ &\boldsymbol{\omega}_{it}^{*\varphi}=[\mathbf{a}_i^{*\varphi}+(1-p(i|\mathbf{x}_t))\mathbf{1}_d/2]\oslash\mathbf{b}_i^{*\varphi}. \end{aligned} \tag{A.6}$$

Update Ξ^*. The gradient $\nabla_{\vartheta}H^{LFA}$ for $\vartheta\in\Xi^*$ is calculated by

$$\begin{aligned} \nabla_{\lambda_i^*}H^{LFA} &\propto \left(\sum_{t=1}^{N}p(i|\mathbf{x}_t)+\xi\lambda_i-1\right)[\Psi'((\xi+N)\lambda_i^*)-\Psi'(\xi+N)], \\ \nabla_{\mathbf{m}_i^*}H^{LFA} &\propto \sum_{t=1}^{N}p(i|\mathbf{x}_t)\boldsymbol{\omega}_{it}^{*\varphi}\odot\boldsymbol{\epsilon}_{it}+\beta(\mathbf{m}_i-\mathbf{m}_i^*), \\ \nabla_{\mathbf{b}_i^{*\nu}}H^{LFA} &\propto \frac{1}{2}\sum_{t=1}^{N}p(i|\mathbf{x}_t)\boldsymbol{\omega}_{it}^{*\nu}\odot\mathrm{diag}(\mathbf{W}_i\mathbf{e}_{it}\mathbf{e}_{it}^T\mathbf{W}_i^T+\Pi_i) \\ &\quad +\mathbf{b}_i^{\nu}\odot(\mathbf{a}_i^{*\nu}\oslash\mathbf{b}_i^{*\nu})-\mathbf{a}_i^{*\nu}+\frac{1}{2}\mathbf{1}_{h_i}, \\ \nabla_{\mathbf{b}_i^{*\varphi}}H^{LFA} &\propto \frac{1}{2}\sum_{t=1}^{N}p(i|\mathbf{x}_t)\boldsymbol{\omega}_{it}^{*\varphi}\oslash\mathbf{b}_i^{*\varphi}\odot\mathrm{diag}(\mathbf{U}_i\Pi_i\mathbf{U}_i^T+\mathbf{I}_d/\beta+\boldsymbol{\epsilon}_{it}\boldsymbol{\epsilon}_{it}^T) \\ &\quad +\mathbf{b}_i^{\varphi}\odot(\mathbf{a}_i^{*\varphi}\oslash\mathbf{b}_i^{*\varphi})-\mathbf{a}_i^{*\varphi}+\frac{1}{2}\mathbf{1}_d. \end{aligned} \tag{A.7}$$

Update: $\{\mathbf{U}_i\}$ The gradient w.r.t. $\mathbf{U}_i$ is calculated by

$$\begin{aligned} \nabla_{\mathbf{U}_i}H^{LFA} &\propto \sum_{t=1}^{N}p(i|\mathbf{x}_t)[\nabla_{\mathbf{U}_i}^{(1)}(i,t)+\delta_{it}\cdot\nabla_{\mathbf{U}_i}^{(2)}(i,t)], \\ \nabla_{\mathbf{U}_i}^{(1)}(i,t) &= \mathbf{W}_i^T\ \mathrm{diag}(\boldsymbol{\omega}_{it}^{*\nu})(\mathbf{W}_i\mathbf{e}_{it}\mathbf{e}_{it}^T\mathbf{W}_i^T+\Pi_i) \\ &\quad +(\mathbf{I}_d-\mathbf{U}_i\mathbf{W}_i)^T\ \mathrm{diag}(\boldsymbol{\omega}_{it}^{*\varphi})\boldsymbol{\epsilon}_{it}\mathbf{e}_{it}^T\mathbf{W}_i^T \\ &\quad -(\mathbf{I}_d-\mathbf{U}_i\mathbf{W}_i)^T\mathbf{D}_i\mathbf{e}_{it}\mathbf{e}_{it}^T\mathbf{W}_i^T\ \mathrm{diag}(\boldsymbol{\omega}_{it}^{*\nu})\Pi_i \\ &\quad -(\mathbf{I}_d-\mathbf{U}_i\mathbf{W}_i)^T[\mathbf{I}_d-\mathbf{D}_i\mathbf{e}_{it}\boldsymbol{\epsilon}_{it}^T]\ \mathrm{diag}(\boldsymbol{\omega}_{it}^{*\varphi})\mathbf{U}_i\Pi_i, \\ \nabla_{\mathbf{U}_i}^{(2)}(i,t) &= 2\ \mathrm{diag}[(\mathbf{a}_i^{\varphi}+\tfrac{1}{2}\mathbf{1}_d)\oslash(\mathbf{b}_i^{\varphi}+\tfrac{1}{2}\ \mathrm{diag}(\mathbf{Q}_{it}))] \\ &\quad \times[(\mathbf{x}_t-\mathbf{U}_i\tilde{\mathbf{y}}_{it}-\tilde{\boldsymbol{\mu}}_{it}^x)\tilde{\mathbf{y}}_{it}^T-\mathbf{U}_i\tilde{\boldsymbol{\Sigma}}_i^{y}], \\ \mathbf{U}_i^{new} &= \mathbb{P}_{GS}(\mathbf{U}_i^{new}) \end{aligned} \tag{A.8}$$

where $\mathbb{P}_{GS}(\cdot)$ denotes the Gram–Schmidt process which outputs an orthogonal matrix, δ_{it} describes the difference between local harmony measure on (i,t) and the weighted average with

$$\delta_{it} = \delta_{it}^{(1)} + \delta_{it}^{(2)},$$

$$\delta_{it}^{(1)} = L_{it}^{(1)} - \sum_{j=1}^{k} p(j|\mathbf{x}_t) L_{jt}^{(1)},$$

$$\begin{aligned} L_{it}^{(1)} = {} & 2\Psi((\xi+N)\lambda_i^*) - 2\Psi(\xi+N) - h_i \ln(2\pi) \\ & + \mathbf{1}_d^T[\Psi(\mathbf{a}_i^{\varphi}) - \ln \mathbf{b}_i^{*\varphi}] + \mathbf{1}_{h_i}^T[\Psi(\mathbf{a}_i^{\nu}) - \ln \mathbf{b}_i^{*\nu}] \\ & - \frac{1}{\beta} \operatorname{tr}[\operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi})] - \mathbf{e}_{it}^T \mathbf{W}_i^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\nu}) \mathbf{W}_i \mathbf{e}_{it} - \boldsymbol{\epsilon}_{it}^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \boldsymbol{\epsilon}_{it} \\ & - \operatorname{tr}[\Pi_i(\mathbf{U}_i^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i + \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\nu}))], \end{aligned}$$

$$\begin{aligned} \delta_{it}^{(2)} = {} & \frac{1}{2} \sum_{\tau=1}^{N} p(i|\mathbf{x}_\tau) \left[L_{i\tau}^{(2)} - \sum_{j=1}^{k} p(j|\mathbf{x}_\tau) L_{j\tau}^{(2)} \right] \\ & - \frac{1}{2} p(i|\mathbf{x}_t) \left[L_{it}^{(2)} - \sum_{j=1}^{k} p(j|\mathbf{x}_t) L_{jt}^{(2)} \right], \end{aligned}$$

$$\begin{aligned} L_{it}^{(2)} = {} & -\operatorname{tr}[\operatorname{diag}(\mathbf{b}_i^{*\nu})^{-1} (\mathbf{W}_i \mathbf{e}_{it} \mathbf{e}_{it}^T \mathbf{W}_i^T + \Pi_i)] \\ & - \operatorname{tr}[\operatorname{diag}(\mathbf{b}_i^{*\varphi})^{-1} (\boldsymbol{\epsilon}_{it} \boldsymbol{\epsilon}_{it}^T + \mathbf{U}_i \Pi_i \mathbf{U}_i^T + \mathbf{I}_d/\beta)]. \end{aligned} \tag{A.9}$$

The gradient for $\vartheta \in \{\boldsymbol{\lambda}, \xi, \{\mathbf{m}_i\}, \beta\}$ is given by

$$\nabla_{\lambda_i} H^{LFA} \propto \sum_{t=1}^{N} p(i|\mathbf{x}_t)\delta_{it} + \xi[\Psi(\xi) - \Psi(\lambda_i \xi) - \Psi(\xi+N) + \Psi((\xi+N)\lambda_i^*)],$$

$$\nabla_{\xi} H^{LFA} \propto \sum_{i=1}^{k} \left[\sum_{t=1}^{N} p(i|\mathbf{x}_t) + \xi\lambda_i - 1 + \Psi(\xi) - \Psi(\lambda_i \xi) - \Psi(\xi+N) + \Psi((\xi+N)\lambda_i^*) \right],$$

$$\nabla_{\mathbf{m}_i} H^{LFA} \propto \begin{cases} \sum\limits_{t=1}^{N} p(i|\mathbf{x}_t)\delta_{it}(\tilde{\mu}_{it}^x - \mathbf{m}_i) + (\mathbf{m}_i^* - \mathbf{m}_i), & \text{each } \mathbf{m}_i \text{ is free,} \\ \sum\limits_{i=1}^{k}\sum\limits_{t=1}^{N} p(i|\mathbf{x}_t)\delta_{it}(\tilde{\mu}_{it}^x - \mathbf{m}_i) + \sum\limits_{i=1}^{k} (\mathbf{m}_i^* - \mathbf{m}_i), & \text{constrain each } \mathbf{m}_i = \mathbf{m}, \end{cases}$$

$$\nabla_{\beta} H^{LFA} \propto \sum_{i=1}^{k} \sum_{t=1}^{N} p(i|\mathbf{x}_t) \nabla_{\beta}(i,t) + dk - \beta \sum_{i=1}^{k} (\mathbf{m}_i^* - \mathbf{m}_i)^T (\mathbf{m}_i^* - \mathbf{m}_i),$$

$$\nabla_{\beta}(i,t) = \frac{1}{\beta} \mathbf{1}_d^T \boldsymbol{\omega}_{it}^{*\varphi} + \delta_{it}[d - \beta \operatorname{tr}[\tilde{\boldsymbol{\Sigma}}_i^{\mu^x} + (\tilde{\boldsymbol{\mu}}_{it}^x - \mathbf{m}_i)(\tilde{\boldsymbol{\mu}}_{it}^x - \mathbf{m}_i)^T]]. \tag{A.10}$$

For hyper-parameters $\{a_{ij}^{\nu}, b_{ij}^{\nu}\}$ of the Gamma prior on $\{\boldsymbol{\nu}_i\}$:

$$\nabla_{\mathbf{a}_i^{\nu}} H^{LFA} \propto \frac{1}{2} \sum_{t=1}^{N} p(i|\mathbf{x}_t) \left[\nabla_{\mathbf{a}_i^{\nu}}^{(1)}(i,t) + \delta_{it} \cdot \nabla_{\mathbf{a}_i^{\nu}}^{(2)}(i,t) \right] + \nabla_{\mathbf{a}_i^{\nu}}^{(3)},$$

$$\begin{aligned} \nabla_{\mathbf{a}_i^{\nu}}^{(1)}(i,t) = {} & -\operatorname{diag}(\mathbf{W}_i \mathbf{e}_{it} \mathbf{e}_{it}^T \mathbf{W}_i^T + \Pi_i) \oslash \mathbf{b}_i^{*\nu} \\ & + 2\boldsymbol{\omega}_{it}^{*\nu} \odot \operatorname{diag}[(\mathbf{W}_i \mathbf{e}_{it} \mathbf{e}_{it}^T \mathbf{W}_i^T + \Pi_i)\Pi_i] \oslash \mathbf{b}_i^{\nu} \\ & - 2 \operatorname{diag}[\mathbf{W}_i \mathbf{e}_{it} \boldsymbol{\epsilon}_{it}^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i \Pi_i] \oslash \mathbf{b}_i^{\nu} \\ & + \operatorname{diag}[\Pi_i \mathbf{U}_i^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i \Pi_i] \oslash \mathbf{b}_i^{\nu} + \Psi'(\mathbf{a}_i^{\nu}), \end{aligned}$$

$$\nabla_{\mathbf{a}_i^{\nu}}^{(2)}(i,t) = \ln \mathbf{b}_i^{\nu} + \Psi(\mathbf{a}_i^{\nu} + \tfrac{1}{2}\mathbf{1}_{h_i}) - \Psi(\mathbf{a}_i^{\nu}) - \ln[\mathbf{b}_i^{\nu} + \tfrac{1}{2} \operatorname{diag}(\tilde{\boldsymbol{\Sigma}}_i^{y} + \tilde{\mathbf{y}}_{it} \tilde{\mathbf{y}}_{it}^T)],$$

$$\nabla_{\mathbf{a}_i^{\nu}}^{(3)} = \ln \mathbf{b}_i^{\nu} - \ln \mathbf{b}_i^{*\nu} + (\mathbf{a}_i^{\nu} - \mathbf{1}_{h_i}) \odot \Psi'(\mathbf{a}_i^{\nu}) - \mathbf{b}_i^{\nu} \oslash \mathbf{b}_i^{*\nu},$$

$$\nabla_{\mathbf{b}_i^{\nu}} H^{LFA} \propto \frac{1}{2} \sum_{t=1}^{N} [\nabla_{\mathbf{b}_i^{\nu}}^{(1)}(i,t) + \delta_{it} \cdot \nabla_{\mathbf{b}_i^{\nu}}^{(2)}(i,t)] + \mathbf{a}_i^{\nu} \oslash \mathbf{b}_i^{\nu} - \sum_{t=1}^{N} \boldsymbol{\omega}_{it}^{*\nu},$$

$$\begin{aligned} \nabla_{\mathbf{b}_i^{\nu}}^{(1)}(i,t) = {} & \operatorname{diag}[(2\mathbf{W}_i \mathbf{e}_{it} \mathbf{e}_{it}^T \mathbf{W}_i^T + \Pi_i) \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\nu}) \Pi_i \\ & - \operatorname{diag}(2\mathbf{W}_i \mathbf{e}_{it} \boldsymbol{\epsilon}_{it}^T - \Pi_i \mathbf{U}_i^T) \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i \Pi_i] \oslash \mathbf{b}_i^{\nu} \odot (\mathbf{a}_i^{\nu} \oslash \mathbf{b}_i^{\nu}), \end{aligned}$$

$$\nabla_{\mathbf{b}_i^{\nu}}^{(2)}(i,t) = \mathbf{a}_i^{\nu} \oslash \mathbf{b}_i^{\nu} - \left(\mathbf{a}_i^{\nu} + \frac{1}{2} \mathbf{1}_{h_i} \right) \oslash \left[\mathbf{b}_i^{\nu} + \frac{1}{2} \operatorname{diag}(\tilde{\boldsymbol{\Sigma}}_i^{y} + \tilde{\mathbf{y}}_{it} \tilde{\mathbf{y}}_{it}^T) \right]. \tag{A.11}$$

For hyper-parameters $\{a_{ij}^{\varphi}, b_{ij}^{\varphi}\}$ of the Gamma prior on $\{\boldsymbol{\varphi}_i\}$, we have

$$\nabla_{\mathbf{a}_i^{\varphi}} H^{LFA} \propto \frac{1}{2} \sum_{t=1}^{N} p(i|\mathbf{x}_t)[\nabla_{\mathbf{a}_i^{\varphi}}^{(1)}(i,t) + \delta_{it} \cdot \nabla_{\mathbf{a}_i^{\varphi}}^{(2)}(i,t)] + \nabla_{\mathbf{a}_i^{\varphi}}^{(3)},$$

$$\begin{aligned} \nabla_{\mathbf{a}_i^{\varphi}}^{(1)}(i,t) = {} & \Psi'(\mathbf{a}_i^{\varphi}) - \operatorname{diag}(\boldsymbol{\epsilon}_{it} \boldsymbol{\epsilon}_{it}^T + \mathbf{U}_i \Pi_i \mathbf{U}_i^T + \mathbf{I}_d/\beta) \oslash \mathbf{b}_i^{*\varphi} \\ & - 2 \operatorname{diag}[(\mathbf{I}_d - \mathbf{U}_i \mathbf{W}_i) \mathbf{e}_{it} (\boldsymbol{\epsilon}_{it}^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i \\ & + (\mathbf{x}_t - \tilde{\boldsymbol{\mu}}_{it}^{y})^T \mathbf{W}_i^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\nu})) \Pi_i \mathbf{U}_i^T] \oslash (\mathbf{b}_i^{\varphi} + \tfrac{1}{2} \operatorname{diag} \tilde{\boldsymbol{\Sigma}}_i^{\mu^y}) \\ & + \operatorname{diag}[\mathbf{U}_i \Pi_i \mathbf{U}_i^T \operatorname{diag}(\mathbf{1}_d + \boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i \Pi_i \mathbf{U}_i^T] \oslash (\mathbf{b}_i^{\varphi} + \tfrac{1}{2} \operatorname{diag} \tilde{\boldsymbol{\Sigma}}_i^{\mu^y}), \end{aligned}$$

$$\nabla_{\mathbf{a}_i^{\varphi}}^{(2)}(i,t) = \ln \mathbf{b}_i^{\varphi} + \Psi(\mathbf{a}_i^{\varphi} + \tfrac{1}{2}\mathbf{1}_d) - \Psi(\mathbf{a}_i^{\varphi}) - \ln[\mathbf{b}_i^{\varphi} + \tfrac{1}{2} \operatorname{diag}(\mathbf{Q}_{it})],$$

$$\nabla_{\mathbf{a}_i^{\varphi}}^{(3)} = \ln \mathbf{b}_i^{\varphi} - \ln \mathbf{b}_i^{*\varphi} + (\mathbf{a}_i^{\varphi} - \mathbf{1}_d) \odot \Psi'(\mathbf{a}_i^{\varphi}) - \mathbf{b}_i^{\varphi} \oslash \mathbf{b}_i^{*\varphi},$$

$$\nabla_{\mathbf{b}_i^{\varphi}} H^{LFA} \propto \frac{1}{2} \sum_{t=1}^{N} p(i|\mathbf{x}_t)[\nabla_{\mathbf{b}_i^{\varphi}}^{(1)}(i,t) + \delta_{it} \cdot \nabla_{\mathbf{b}_i^{\varphi}}^{(2)}] + \mathbf{a}_i^{\varphi} \oslash \mathbf{b}_i^{\varphi} - \sum_{t=1}^{N} \boldsymbol{\omega}_{it}^{*\varphi},$$

$$\begin{aligned} \nabla_{\mathbf{b}_i^{\varphi}}^{(1)}(i,t) = {} & \operatorname{diag}[2(\mathbf{I}_d - \mathbf{U}_i \mathbf{W}_i) \mathbf{e}_{it} \mathbf{e}_{it}^T \mathbf{W}_i^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\nu}) \mathbf{W}_i \\ & - 2(\mathbf{I}_d - \mathbf{U}_i \mathbf{W}_i) \mathbf{e}_{it} \boldsymbol{\epsilon}_{it}^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i \mathbf{W}_i \\ & - \mathbf{U}_i \Pi_i (\operatorname{diag}(\boldsymbol{\omega}_{it}^{*\nu}) + \mathbf{U}_i^T \operatorname{diag}(\boldsymbol{\omega}_{it}^{*\varphi}) \mathbf{U}_i) \mathbf{W}_i] \\ & \oslash (\mathbf{b}_i^{\varphi} + \tfrac{1}{2} \operatorname{diag}(\tilde{\boldsymbol{\Sigma}}_i^{\mu^y})), \end{aligned}$$

$$\nabla_{\mathbf{b}_i^{\varphi}}^{(2)}(i,t) = \mathbf{a}_i^{\varphi} \oslash \mathbf{b}_i^{\varphi} - (\mathbf{a}_i^{\varphi} + \tfrac{1}{2}\mathbf{1}_d) \oslash [\mathbf{b}_i^{\varphi} + \tfrac{1}{2} \operatorname{diag}(\mathbf{Q}_{it})]. \tag{A.12}$$

Table B1
VB algorithm to maximize the variational lower bound in Eq. (23) on LFA with DNG prior (VB-LFA).

```
Initialization: Randomly initialize the model with large enough number $k$ of components and hidden dimensionalities $\{h_i\}$; set $\tau = 0$ and the variational function $J_{VB}(\tau) = -\infty$;
repeat
    E-step: Given $\Xi$ and $\Xi^*$, for each $i$ and $t$, estimate $p_{it}$, $\overline{\mathbf{y}}_{it}^*$ and $\boldsymbol{\Sigma}_i^{*y}$ by Eq. (B.1).
    M-step: Given $p_{it}$, $\overline{\mathbf{y}}_{it}^*, \boldsymbol{\Sigma}_i^{*y}$ and $\Xi$, update $\Xi^*$ and $\{\mathbf{U}_i\}$ by Eq. (B.2);
    H-step: Given $p_{it}$, $\overline{\mathbf{y}}_{it}^*, \boldsymbol{\Sigma}_i^{*y}$ and $\Xi^*$, update prior hyper-parameters $\Xi$ by Eq. (B.3).
    $\otimes$ for $i = 1, \dots, k$ do
        if $\lambda_i^*$ or $\lambda_i \to 0$ then discard component $i$, let $k = k - 1$;
        for $j = 1, \dots, h_i$ do
            if $\frac{b_{ij}^{*\nu}}{a_{ij}^{*\nu}}$ or $\frac{b_{ij}^{\nu}}{a_{ij}^{\nu}} \to 0$ then discard dimension $j$ in component $i$, let $h_i = h_i - 1$;
    if another 5 runs pass then let $\tau = \tau + 1$; calculate $J_{VB}(\tau)$ by Eq. (23);
until $J_{VB}(\tau) - J_{VB}(\tau - 1) < \epsilon J_{VB}(\tau - 1)$, with $\epsilon = 10^{-5}$ in our implementation
```

Appendix B. The VB-LFA algorithm

E-step update p_{it}, $\bar{\mathbf{y}}^*_{it}$ and $\{\mathbf{\Sigma}_i^{*y}\}$:

$$p_{it} = r_{it}^{1/2} \Big/ \left(\sum_{j=1}^{k} r_{jt}^{1/2} \right).,$$

$$r_{it} = \exp\left\{ 2\Psi(\xi^*\lambda_i^*) + h_i + \sum_{j=1}^{h_i} [\Psi(a_{ij}^{*\nu}) - \ln b_{ij}^{*\nu}] + \sum_{j=1}^{d} [\Psi(a_{ij}^{*\varphi}) - \ln b_{ij}^{*\varphi}] \right.$$
$$- \mathrm{tr}[\mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi})[(\mathbf{x}_t - \mathbf{U}_i\bar{\mathbf{y}}^*_{it} - \mathbf{m}_i^*)(\mathbf{x}_t - \mathbf{U}_i\bar{\mathbf{y}}^*_{it} - \mathbf{m}_i^*)^T$$
$$\left. + \mathbf{\Sigma}_i^{*\mu} + \mathbf{U}_i\mathbf{\Sigma}_i^{*y}\mathbf{U}_i]] - \mathrm{tr}[\mathrm{diag}(\mathbf{a}_i^{*\nu} \oslash \mathbf{b}_i^{*\nu})(\bar{\mathbf{y}}^*_{it}\bar{\mathbf{y}}^{*T}_{it} + \mathbf{\Sigma}_i^{*y})] \right\},$$

$$\bar{\mathbf{y}}^*_{it} = [\mathbf{U}_i^T \ \mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi})\mathbf{U}_i + \mathrm{diag}(\mathbf{a}_i^{*\nu} \oslash \mathbf{b}_i^{*\nu})]^{-1}\mathbf{U}_i^T \ \mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi})(\mathbf{x}_t - \mathbf{m}_i^*),$$

$$\mathbf{\Sigma}_i^{*y} = [\mathbf{U}_i^T \ \mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi})\mathbf{U}_i + \mathrm{diag}(\mathbf{a}_i^{*\nu} \oslash \mathbf{b}_i^{*\nu})]^{-1}. \qquad \text{(B.1)}$$

M-step update posterior hyper-parameters $\mathbf{\Xi}^* = \{\lambda^*, \xi^*, \{\mathbf{m}_i^*, \mathbf{\Sigma}_i^{*\mu}\}, \{a_{ij}^{*\nu}, b_{ij}^{*\nu}\}, \{a_{ij}^{*\varphi}, b_{ij}^{*\varphi}\}\}$ and $\{\mathbf{U}_i\}$: (Table B1)

$$\lambda_i^* = \left(\xi\lambda_i + \sum_{t=1}^{N} p_{it} \right) / (\xi + N), \quad \xi^* = \xi + N,$$

$$\mathbf{m}_i^* = \left[\beta\mathbf{I}_d + \left(\sum_{t=1}^{N} p_{it} \right) \mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi}) \right]^{-1}$$
$$\times \left[\beta\mathbf{m}_i + \sum_{t=1}^{N} p_{it} \ \mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi})(\mathbf{x}_t - \mathbf{U}_i\bar{\mathbf{y}}^*_{it}) \right],$$

$$\mathbf{\Sigma}_i^{*\mu} = \left[\beta\mathbf{I}_d + \left(\sum_{t=1}^{N} p_{it} \right) \mathrm{diag}(\mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi}) \right]^{-1},$$

$$\mathbf{a}_i^{*\nu} = \mathbf{a}_i^{\nu} + \frac{1}{2} \sum_{t=1}^{N} p_{it} \mathbf{1}_{h_i},$$

$$\mathbf{b}_i^{*\nu} = \mathbf{b}_i^{\nu} + \frac{1}{2} \sum_{t=1}^{N} p_{it} \ \mathrm{diag}[\bar{\mathbf{y}}^*_{it}\bar{\mathbf{y}}^{*T}_{it} + \mathbf{\Sigma}_i^{*y}],$$

$$\mathbf{a}_i^{*\varphi} = \mathbf{a}_i^{\varphi} + \frac{1}{2} \sum_{t=1}^{N} p_{it} \mathbf{1}_d,$$

$$\mathbf{b}_i^{*\varphi} = \mathbf{b}_i^{\varphi} + \frac{1}{2} \sum_{t=1}^{N} p_{it} \ \mathrm{diag}[(\mathbf{x}_t - \mathbf{U}_i\bar{\mathbf{y}}^*_{it} - \mathbf{m}_i^*)(\mathbf{x}_t - \mathbf{U}_i\bar{\mathbf{y}}^*_{it} - \mathbf{m}_i^*)^T + \mathbf{\Sigma}_i^{*\mu} + \mathbf{U}_i\mathbf{\Sigma}_i^{*y}\mathbf{U}_i^T],$$

$$\mathbf{U}_i^{new} = \mathbb{P}_{GS}(\mathbf{U}_i^{new}), \mathbf{U}_i^{new} = \mathbf{U}_i^{old} + \eta(\mathbf{G}_{U_i} - \mathbf{U}_i^{old}\mathbf{G}_{U_i}^T\mathbf{U}_i^{old}),$$

$$\mathbf{G}_{U_i} = \left[\sum_{t=1}^{N} p_{it}(\mathbf{x}_t - \mathbf{m}_i^*)\bar{\mathbf{y}}^{*T}_{it} \right]$$
$$\left(\sum_{t=1}^{N} p_{it}\bar{\mathbf{y}}^*_{it}\bar{\mathbf{y}}^{*T}_{it} + \mathbf{\Sigma}_i^{*y} \right)^{-1} - \mathbf{U}_i^{old}. \qquad \text{(B.2)}$$

H-step update prior hyper-parameters $\mathbf{\Xi} = \{\lambda, \xi, \{\mathbf{m}_i\}, \beta, \{a_{ij}^{\nu}, b_{ij}^{\nu}\}, \{a_{ij}^{\varphi}, b_{ij}^{\varphi}\}\}$:

$$\lambda_i^{new} = (\lambda_i^{old} + \eta\delta\lambda_i) \Big/ \left[\sum_{j=1}^{k} (\lambda_j^{old} + \eta\delta\lambda_j) \right].,$$
$$\delta\lambda_i = \Psi(\xi^*\lambda_i^*) - \Psi(\xi^*) - \Psi(\xi\lambda_i) + \Psi(\xi),$$

$$\xi^{new} = \xi^{old} + \eta\delta\xi, \quad \delta\xi = \sum_{i=1}^{k} \lambda_i^{old}\delta\lambda_i,$$

$$\mathbf{m}_i = \begin{cases} \mathbf{m}_i^*, & \text{general case (each } \mathbf{m}_i \text{ is free)}, \\ \dfrac{1}{k} \sum_{i=1}^{k} \mathbf{m}_i^*, & \text{special case (constrain each } \mathbf{m}_i = \mathbf{m}), \end{cases}$$

$$\beta = kd \Big/ \left\{ \sum_{i=1}^{k} [(\mathbf{m}_i - \mathbf{m}_i^*)(\mathbf{m}_i - \mathbf{m}_i^*)^T + \mathrm{tr}(\mathbf{\Sigma}_i^{*\mu})] \right\}.,$$

$$\mathbf{a}_i^{\nu} new = \mathbf{a}_i^{\nu} old + \eta\delta(\mathbf{a}_i^{\nu}), \quad \delta(\mathbf{a}_i^{\nu}) = \ln \mathbf{b}_i^{\nu} - \ln \mathbf{b}_i^{*\nu} - \Psi(\mathbf{a}_i^{\nu}) + \Psi(\mathbf{a}_i^{*\nu}),$$

$$\mathbf{b}_i^{\nu} new = \mathbf{b}_i^{\nu} old + \eta\delta(\mathbf{b}_i^{\nu}), \quad \delta(\mathbf{b}_i^{\nu}) = \mathbf{a}_i^{\nu} \oslash \mathbf{b}_i^{\nu} - \mathbf{a}_i^{*\nu} \oslash \mathbf{b}_i^{*\nu},$$

$$\mathbf{a}_i^{\varphi} new = \mathbf{a}_i^{\varphi} old + \eta\delta(\mathbf{a}_i^{\varphi}), \quad \delta(\mathbf{a}_i^{\varphi}) = \ln \mathbf{b}_i^{\varphi} - \ln \mathbf{b}_i^{*\varphi} - \Psi(\mathbf{a}_i^{\varphi}) + \Psi(\mathbf{a}_i^{*\varphi}),$$

$$\mathbf{b}_i^{\varphi} new = \mathbf{b}_i^{\varphi} old + \eta\delta(\mathbf{b}_i^{\varphi}), \quad \delta(\mathbf{b}_i^{\varphi}) = \mathbf{a}_i^{\varphi} \oslash \mathbf{b}_i^{\varphi} - \mathbf{a}_i^{*\varphi} \oslash \mathbf{b}_i^{*\varphi}. \qquad \text{(B.3)}$$

References

[1] R. Redner, H. Walker, Mixture densities, maximum likelihood and the EM algorithm, SIAM Rev. 26 (2) (1984) 195–239.

[2] G.E. Hinton, M. Revow, P. Dayan, Recognizing handwritten digits using mixtures of linear models, in: NIPS, 1994, pp. 1015–1022.

[3] M.A.F. Figueiredo, A.K. Jain, Unsupervised learning of finite mixture models, IEEE Trans. Pattern Anal. Mach. Intell. 24 (3) (2002) 381–396.

[4] L. Shi, S. Tu, L. Xu, Learning Gaussian mixture with automatic model selection: a comparative study on three Bayesian related approaches, a special issue on machine learning and intelligence science: IScIDE2010 (B), J. Front. Elect. Electron. Eng. China 6 (2) (2011) 215–244.

[5] D. Rubin, D. Thayer, EM algorithms for ML factor analysis, Psychometrika 47 (1) (1982) 69–76.

[6] Z. Ghahramani, M.J. Beal, Variational inference for Bayesian mixtures of factor analysers, in: NIPS, 1999, pp. 449–455.

[7] L. Xu, Advances on BYY harmony learning: information theoretic perspective, generalized projection geometry, and independent factor auto-determination, IEEE Trans. Neural Netw. 15 (5) (2004) 885–902.

[8] L. Xu, Bayesian Ying Yang system, best harmony learning and Gaussian manifold based family, in: J. Zurada, G. Yen, J. Wang (Eds.), Computational Intelligence: Research Frontiers, WCCI 2008 Plenary/Invited Lectures, Lecture Notes in Computer Science, vol. 5050, Springer-Verlag, Berlin-Heidelberg, 2008, pp. 48–78.

[9] S. Tu, L. Xu, Parameterizations make different model selections: empirical findings from factor analysis, a special issue on machine learning and intelligence science: IScIDE2010 (B), J. Front. Electr. Electron. Eng. China 6 (2) (2011) 256–274.

[10] Z. Ghahramani, G. Hinton, The EM Algorithm for Mixtures of Factor Analyzers, Technical Report CRG-TR-96-1, Department of Computer Science, University of Toronto, Toronto, Canada, 1997.

[11] M.E. Tipping, C.M. Bishop, Mixtures of probabilistic principal component analyzers, Neural Comput. 11 (2) (1999) 443–482.

[12] H. Akaike, A new look at the statistical model identification, IEEE Trans. Autom. Control 19 (6) (1974) 716–723.

[13] L. Xu, Bayesian Ying-Yang system, best harmony learning, and five action circling, a special issue on emerging themes on information theory and Bayesian Approach, Front. Electr. Electron. Eng. China 5 (3) (2010) 281–328.

[14] A.A. Salah, E. Alpaydin, Incremental mixtures of factor analysers, in: Proceedings of the ICPR'2003, vol. 1, 2004, pp. 276–279.

[15] L. Xu, On essential topics of BYY harmony learning: current status, challenging issues, and gene analysis applications, a special issue on machine learning and intelligence science: IScIDE2010 (C), J. Front. Electr. Electron. Eng. China 7 (1) (2012) 147–196.

[16] L. Xu, A. Krzyzak, E. Oja, Unsupervised and supervised classifications by rival penalized competitive learning, in: Proceedings of ICPR'92, vol. II, Hauge, Netherlands, 1992, pp. 496–499.

[17] L. Xu, A. Krzyzak, E. Oja, Rival penalized competitive learning for clustering analysis, RBF net and curve detection, IEEE Trans. Neural Networks 4 (4) (1993) 636–649.

[18] L. Xu, Bayesian-Kullback coupled Ying-Yang machines: unified learning and new results on vector quantization, in: Proceedings of International Conference on Neural Information Processing, Beijing, China, 1995, pp. 977–988, (A further version in NIPS8, D.S. Touretzky, et al. (Eds.), MIT Press, 444–450).

[19] L. Xu, Bayesian Ying Yang learning, in: Scholarpedia 2(3) 1809, ⟨http://scholarpedia.org/article/Bayesian_Ying_Yang_Learning⟩, 2007.

[20] L. Xu, Codimensional matrix pairing perspective of BYY harmony learning: hierarchy of bilinear systems, joint decomposition of data-covariance, and applications of network biology, a special issue on machine learning and intelligence science: IScIDE2010 (A), J. Front. Electr. Electron. Eng. China 6 (1) (2011) 86–119.

[21] R. Neal, G.E. Hinton, A view of the EM algorithm that justifies incremental, sparse, and other variants, in: Learning in Graphical Models, Kluwer Academic Publishers, 1998, pp. 355–368.

[22] J. Rissanen, Basics of estimation, Front. Electr. Electron. Eng. China 5 (3) (2010) 274–280.

[23] L. Xu, Bayesian Ying-Yang learning theory for data dimension reduction and determination, J. Comput. Intell. Finance 6 (5) (1998) 6–18.

[24] S. Kotz, S. Nadarajah, Multivariate t Distributions and Their Applications, Cambridge University Press, Cambridge, 2004.

[25] L. Shi, Automatic Model Selection on Local Gaussian Structures with Priors: Comparative Investigations and Applications (Ph.D. thesis). The Chinese University of Hong Kong (in preparation) (2012).

[26] W.M. Rand, Objective criteria for the evaluation of clustering methods, J. Am. Stat. Assoc. 66 (336) (1971) 846–850.

[27] W. Xu, X. Liu, Y. Gong, Document clustering based on nonnegative matrix factorization, in: Proceedings of SIGIR'03, 2003, pp. 267–273.

[28] X.V. Nguyen, J. Epps, J. Bailey, Information theoretic measures for clusterings comparison: variants, properties, normalization and correction for chance, J. Mach. Learn. Res. 11 (2010) 2837–2854.

[29] M. Varma, A. Zisserman, Texture classification: are filter banks necessary?, in: Proceedings of CVPR'03, vol. 2, 2003, pp. 691–698.

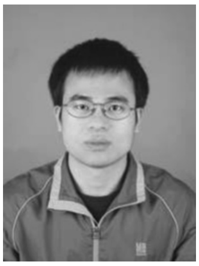
[30] C. Nikou, A. Likas, N. Galatsanos, A Bayesian framework for image segmentation with spatially varying mixtures, IEEE Trans. Image Process. 19 (9) (2010) 2278–2289.

[31] R. Unnikrishnan, C. Pantofaru, M. Hebert, Toward objective evaluation of image segmentation algorithms, IEEE Trans. Pattern Anal. Mach. Intell. 29 (6) (2007) 929–944.

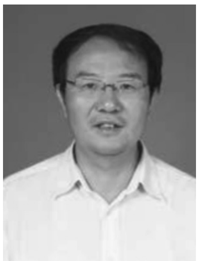
[32] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

Lei Shi obtained his B.Eng. degree in Computer Science and Technology from University of Science and Technology of China, in 2005. He is currently a Ph.D student of Department of Computer Science and Engineering in The Chinese University of Hong Kong. His research interests include statistical learning and neural computing.

Zhi-Yong Liu is an associate professor at the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His research interests include machine learning, pattern recognition, computer vision, and bioinformatics.



Shikui Tu is a Ph.D candidate of the Department of Computer Science and Engineering, The Chinese University of HongKong. He obtained his Bachelor degree from School of Mathematical Science, Peking University, in 2006. His research interests include statistical learning, pattern recognition, and bioinformatics.



Lei Xu is a chair professor at the Department of Computer Science and Engineering, Chinese University of Hong Kong, Statin, Hong Kong. His research interests include statistical learning, computer vision, and bioinformatics.