ISSN 1673-3460

Volume 6 · Number 2 · June 2011

Available online

www.springerlink.com

Frontiers of Electrical and Electronic Engineering in China

高等教育出版社
HIGHER EDUCATION PRESS

Springer

Front. Electr. Electron. Eng. China 2011, 6(2): 256–274
DOI 10.1007/s11460-011-0150-2

RESEARCH ARTICLE

Shikui TU, Lei XU

Parameterizations make different model selections: Empirical findings from factor analysis

Abstract How parameterizations affect model selection performance is an issue that has been ignored or seldom studied since traditional model selection criteria, such as Akaike's information criterion (AIC), Schwarz's Bayesian information criterion (BIC), difference of negative log-likelihood (DNLL), etc., perform equivalently on different parameterizations that have equivalent likelihood functions. For factor analysis (FA), in addition to one traditional model (shortly denoted by FA-a), it was previously found that there is another parameterization (shortly denoted by FA-b) and the Bayesian Ying-Yang (BYY) harmony learning gets different model selection performances on FA-a and FA-b. This paper investigates a family of FA parameterizations that have equivalent likelihood functions, where each one (shortly denoted by FA-r) is featured by an integer r, with FA-a as one end that $r = 0$ and FA-b as the other end that r reaches its upper-bound. In addition to the BYY learning in comparison with AIC, BIC, and DNLL, we also implement variational Bayes (VB). Several empirical finds have been obtained via extensive experiments. First, both BYY and VB perform obviously better on FA-b than on FA-a, and this superiority of FA-b is reliable and robust. Second, both BYY and VB outperform AIC, BIC, and DNLL, while BYY further outperforms VB considerably, especially on FA-b. Moreover, with FA-a replaced by FA-b, the gain obtained by BYY is obviously higher than the one by VB, while the gain by VB is better than no gain by AIC, BIC, and DNLL. Third, this paper also demonstrates how each part of priors incrementally and jointly improves the performances, and further shows that using VB to optimize the hyperparameters of priors deteriorates the performances while using BYY for this purpose can further improve the performances.

Received March 23, 2011; accepted April 20, 2011

Shikui TU, Lei XU (✉)
Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China
E-mail: lxu@cse.cuhk.edu.hk

Keywords model selection, factor analysis, parameterizations, maximum likelihood, variational Bayes, Bayesian Ying-Yang learning

1 Introduction

Model selection is traditionally implemented in two stages. The first stage enumerates a set of candidate models via an index k that represents the complexity of the corresponding model and estimates the parameter $\hat{\theta}_k$ that maximizes the likelihood $L(\theta_k)$, while the second stage selects a best complexity k according to one of typical criteria, such as Akaike's information criterion (AIC) [1], Schwarz's Bayesian information criterion (BIC) [2], Rissanen's minimum description length (MDL) [3] (which stems from another viewpoint but coincides with BIC when it is simplified to a simple computable criterion), in a format of

$$\mathcal{J}(k) = L(\hat{\theta}_k) + C(k). \tag{1}$$

Two candidate models with different parameterizations have the same model selection performance if they share equivalent likelihoods and the complexity term $C(k)$. Consequently, how parameterizations affect model selection performance was an issue that has been ignored or seldom studied.

Factor analysis (FA) [4] models the observed multidimensional vector with the help of a low-dimensional Gaussian latent vector (or factors) through a linear transform by a factor loading matrix, plus a Gaussian noise vector. It is usually used as a linear technique of dimensionality reduction [5,6]. Moreover, the maximum likelihood solution of FA with an isotropic noise covariance matrix extracts principal components of the observed data [4,7]. Traditionally, FA is made on a parameterization that takes the form of a free factor loading matrix and a unit covariance matrix for the latent factors, which has been widely used in various studies, e.g., in Refs. [8–10]. For simplicity, we shortly denote this

parameterization as FA-a.

In the literature, the parameterization issue of a statistical model has been studied within Bayesian paradigm on the performance of numerical techniques in making inferences rather than model selection. Reparameterization techniques include parameter transformation to posterior normality and orthogonality [11,12], data augmentation (adding latent variables) and parameter expansion (adding new parameters) to improve the computational accuracy and efficiency, such as fitting the data more accurately, speeding up the Gibbs sampler for posterior [13], and so on. Recently, FA-a is overparameterized in Ref. [14] to obtain a fast Gibbs sampler for a posterior distribution, where the factor loading matrix has a lower triangular structure and the covariance matrix for the latent factors is diagonal.

In the Item 9.4 of Ref. [15], an alternative FA parametrization has been proposed and implemented by the Bayesian Ying-Yang (BYY) harmony learning, which constrains the factor loading matrix to be a rectangular orthogonal matrix, and allows free parameters as the diagonal covariance matrix of the latent variables. Here, we shortly denote this parameterization as FA-b. FA-a and FA-b are equivalent by the maximum likelihood (ML) learning because the corresponding two likelihood functions are equivalent, and thus get the same performance of model selection under the criterion of Eq. (1). However, it was found that the BYY harmony learning gets different model selection performances on FA-a and FA-b [16–19].

This paper continues the above study to further examine how parameterizations affect model selection performance. We combine FA-a and FA-b into a family of FA parameterizations that have equivalent likelihood functions. Each instance in this family is featured by an integer r and thus shortly denoted by FA-r, with FA-a as one end that $r = 0$ and FA-b as the other end that r reaches its upper-bound m. Between the two ends, FA-r is a mixture of an r hidden factor based FA-b and an $m - r$ hidden factor based FA-a, with r indicating the number of free parameters in the diagonal covariance matrix of the hidden variables. This paper aims at a systematic empirical investigation on this family of parameterizations. In addition to the BYY learning, we take in consideration not only variational Bayes (VB) that has been popularly studied in Refs. [20,21], but also AIC, BIC, and difference of negative log-likelihood (DNLL) in a format of Eq. (1). Moreover, we also take in consideration how each part of priors incrementally and jointly improves the performances of BYY and VB, and whether optimizing the hyper-parameters within these priors can further improve the performances.

Several empirical finds have been obtained via extensive experiments. First, both BYY and VB perform better on FA-b than on FA-a. Specifically, both BYY and VB reach their best performances on one parameterization FA-m^* with m^* being the correct number of hidden factors. This provides a correct calibration though m^* is unknown. On one hand, the performances on those of FA-r drop sharply as r reduces from m^* towards to FA-a, which means that the contribution of FA-a is negative. On the other hand, the performance of FA-r reduces slightly and slowly as r increases towards to FA-b. Moreover, we make a comparison on FA-b with its initial dimension set at r and found a performance similar to that on FA-b. Therefore, FA-b is superior to FA-a considerably and reliably. Second, both BYY and VB outperform AIC, BIC, and DNLL, while BYY further outperforms VB, especially on FA-b. Moreover, with FA-a replaced by FA-b, the gain obtained by BYY is obviously higher than the one by VB, while the gain by VB is better than no gain by AIC, BIC, and DNLL, especially for a finite size of samples. Third, we also provide a systematic investigation on how each part of the priors contributes to the model selection performance, and find that though the performance of either VB or BYY can be improved with the help of appropriate priors, BYY does not highly depend on the presences of the priors whereas VB does. Moreover, optimizing the hyper-parameters of priors by BYY further improves the performances while using VB for this purpose actually deteriorates the performances.

The rest of this paper is organized as follows. Section 2 introduces FA-a and a two-stage procedure for the hidden dimensionality estimation problem. Section 3 presents the ML equivalent family of parameterizations, FA-r. Section 4 is devoted to VB learning on FA-r. Though there is one algorithm available (see Sect. 3.2 in Ref. [22]) for making the BYY learning, no study has been made with appropriate priors added and the hyper-parameters of theses priors updated, for which we derive the learning algorithms in Sect. 5. Then, Sect. 6 gives a systematic empirical analysis on all learning algorithms based on FA-r. Section 7 concludes this paper.

2 FA and its modeling task

FA is a statistical method that models the observed random variables as linear combinations of fewer hidden variables (called factors) plus some noise, i.e.,

$$\boldsymbol{x} = \boldsymbol{A}\boldsymbol{y} + \boldsymbol{\mu} + \boldsymbol{e}, \tag{2}$$

where $\boldsymbol{x}$ is an n-dimensional observation, $\boldsymbol{y}$ is an $m \times 1$ hidden factor vector, $\boldsymbol{A}$ is an $n \times m$ factor loading matrix, $\boldsymbol{\mu}$ is an unknown constant, and $\boldsymbol{e}$ is an $n \times 1$ noise vector. Moreover, $\boldsymbol{y}$ and $\boldsymbol{e}$ are assumed to be Gaussian distributed, and uncorrelated, i.e., $\mathrm{E}[\boldsymbol{y}\boldsymbol{e}^{\mathrm{T}}] = \boldsymbol{0}$. Usually, $m < n$, where m is the number of hidden factors or the hidden dimensionality.

We further specify the mathematical details of the joint likelihood $q(\boldsymbol{x}, \boldsymbol{y}|\boldsymbol{\Theta}_m)$ with $\boldsymbol{\Theta}_m$ representing all unknown parameters. One conventional parameterization is the FA-a given in Table 1, where the factor loading is an arbitrary matrix and the latent factors' covariance matrix is an identity matrix. FA-a is commonly used in statistics [4] and machine learning [7,9,23].

Given an independent and identically distributed (i.i.d.) sample set $X_N = \{\boldsymbol{x}_t\}_{t=1}^N$, the task of FA modeling consists of three levels of inverse problems (see Sect. 1.1 in Ref. [22]), i.e., inferring $\{\boldsymbol{y}_t\}$, learning parameters $\boldsymbol{\Theta}_m$ and selecting an appropriate m. The three problems are sequentially nested within a hierarchy. Usually, parameters are estimated under the ML principle:

$$\begin{aligned} \hat{\boldsymbol{\Theta}}_m^{\mathrm{ML}} &= \arg\max_{\boldsymbol{\Theta}_m} \ln q(X_N|\boldsymbol{\Theta}_m), \\ q(X_N|\boldsymbol{\Theta}_m) &= \prod_t q(\boldsymbol{x}_t|\boldsymbol{\Theta}_m), \end{aligned} \tag{3}$$

which is implemented by an expectation-maximization (EM) algorithm [7,24].

Selecting an appropriate hidden dimensionality is a model selection problem, conventionally tackled by a two-stage procedure, i.e., at Stage I parameter learning is repeated on a set of candidate hidden dimensionalities among which one is selected via a criterion at Stage II. Classical model selection criteria include AIC [1,25] and BIC/MDL [2,3]. The FA hidden dimensionality is determined as $\hat{m}$:

$$\begin{aligned} &\hat{m} = \arg\min_{m\in\mathcal{M}} \mathcal{J}_{\mathrm{Cri}}, \\ &\mathcal{J}_{\mathrm{Cri}} = \begin{cases} -\ln q(X_N|\hat{\boldsymbol{\Theta}}_m^{\mathrm{ML}}) + d_m, & \text{AIC}, \\ -\ln q(X_N|\hat{\boldsymbol{\Theta}}_m^{\mathrm{ML}}) + \frac{\ln N}{2} d_m, & \text{BIC/MDL}, \end{cases} \end{aligned} \tag{4}$$

where $\mathcal{M}$ is a set of candidate hidden dimensionalities, and d_m is the number of degrees of freedom in FA. Equations (3) and (4) constitute the conventional *two-stage procedure*. Equation (4) provides two specific examples of Eq. (1). Another choice for $\mathcal{J}_{\mathrm{Cri}}$ is the logarithm of the likelihood-ratio or the difference of negative log-likelihood (DNLL):

$$\mathcal{J}_{\mathrm{DNLL}} = -\ln q(X_N|\hat{\boldsymbol{\Theta}}_m^{\mathrm{ML}}) + \ln q(X_N|\hat{\boldsymbol{\Theta}}_{m-1}^{\mathrm{ML}}), \tag{5}$$

which allows model selection by capturing the decrement of the negative log-likelihood as the candidate hidden dimensionality increases by one.

In the following, we consider FA with $\boldsymbol{\Sigma}_e = \sigma_e^2 \boldsymbol{I}_n$, which leads FA equivalent to principal component analysis (PCA) [4,7] under the ML principle. Without loss of generality, we also assume $\boldsymbol{\mu} = \boldsymbol{0}$.

3 ML-equivalent parameterizations of FA

In this paper, the ML-equivalence between two FA parameterizations means "the corresponding two likelihood functions are equivalent". The FA-b [15–17], listed in Table 1, is another parameterization for FA, and it is ML-equivalent to FA-a because we have $G(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\Sigma}_x)$, with $\boldsymbol{\Sigma}_x = \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{\Sigma}_e$ for FA-a and $\boldsymbol{\Sigma}_x = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}} + \boldsymbol{\Sigma}_e$ for FA-b. It can be observed that the ML estimation seeks a positive definite matrix $\boldsymbol{\Sigma}_x$ or its equivalent decomposition into either of $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}}$ and $\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}$.

Since FA-a has more number of free parameters than FA-b, they are different under AIC or BIC by Eq. (4) if the number of free parameters is directly used as d_m. In practice [4,7,17], the extra degrees of freedom in FA-a are actually subtracted, i.e., $d_m = nm + 1 - \frac{m(m-1)}{2}$, equal to the number of free parameters in FA-b. Thus, we get the same $\hat{m}$ under AIC or BIC by Eq. (4).

Although FA-a and FA-b are equivalent in model selection under AIC or BIC, they have been pointed out to be different under the BYY learning in Refs. [15,16]. This motivates us to further investigate how the forms of parameterizations affect model selection performance. For a systematic study, we present a new family of ML-equivalent FA parameterizations varying from FA-a to FA-b as follows.

The difference between FA-a and FA-b mainly comes from how to encode the hidden variable $\boldsymbol{y}$'s complexity. Following this nature, we construct the following FA model:

$$\begin{aligned} &\boldsymbol{x} = \boldsymbol{V}_r \boldsymbol{y} + \boldsymbol{\mu} + \boldsymbol{e},\ \boldsymbol{V}_r = [\boldsymbol{U}_r, \boldsymbol{A}_{m-r}], \\ &\boldsymbol{y} \text{ comes from } G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Sigma}_y^r), \\ &\boldsymbol{\Sigma}_y^r = \mathrm{diag}[\nu_1^{-1}, \ldots, \nu_r^{-1}, 1, \ldots, 1], \end{aligned} \tag{6}$$

Table 1 Two probabilistic parameterizations of FA, namely FA-a and FA-b (E[·] denotes the *expectation*, and $G(\bullet|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes a Gaussian distribution with the mean vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$, and $\mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_m]$ is a diagonal matrix with $\lambda_1, \lambda_2, \ldots, \lambda_m$ as its diagonal elements. $\boldsymbol{I}_m$ is an $m \times m$ identity matrix. $\boldsymbol{\Sigma}_e$ is a diagonal positive definite matrix. Here and throughout this paper, both $q(\cdot)$ and $p(\cdot)$ denote probability distributions.)

	type-A FA-a: $\boldsymbol{\Theta}_m^a = \{\boldsymbol{A}, \boldsymbol{\mu}, \boldsymbol{\Sigma}_e\}$	type-B FA-b: $\boldsymbol{\Theta}_m^b = \{\boldsymbol{U}, \boldsymbol{\mu}, \boldsymbol{\Lambda}, \boldsymbol{\Sigma}_e\}$
$\mathrm{E}[\boldsymbol{y}\boldsymbol{e}^{\mathrm{T}}]$	$\boldsymbol{0}$ ($\boldsymbol{y}$ and $\boldsymbol{e}$ uncorrelated)	$\boldsymbol{0}$ ($\boldsymbol{y}$ and $\boldsymbol{e}$ uncorrelated)
$q(\boldsymbol{y}|\boldsymbol{\Theta})$	$G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{I}_m)$	$G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Lambda})$, $\boldsymbol{\Lambda} = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_m]$
$\boldsymbol{A}$	any full column rank matrix	$\boldsymbol{A} = \boldsymbol{U}$, $\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = \boldsymbol{I}_m$
$q(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\Theta})$	$G(\boldsymbol{x}|\boldsymbol{A}\boldsymbol{y} + \boldsymbol{\mu}, \boldsymbol{\Sigma}_e)$	$G(\boldsymbol{x}|\boldsymbol{U}\boldsymbol{y} + \boldsymbol{\mu}, \boldsymbol{\Sigma}_e)$
$q(\boldsymbol{x}|\boldsymbol{\Theta})$	$G(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{A}\boldsymbol{A}^{\mathrm{T}} + \boldsymbol{\Sigma}_e)$	$G(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}} + \boldsymbol{\Sigma}_e)$

where $\boldsymbol{\Sigma}_y^r$ is $\boldsymbol{y}$'s covariance matrix with $m-r$ constant 1s in the diagonal. Moreover, we have $\boldsymbol{U}_r \in \mathbb{R}^{n\times r}$, $\boldsymbol{U}_r^{\mathrm{T}}\boldsymbol{U}_r = \boldsymbol{I}_r$, $\boldsymbol{A}_{m-r} \in \mathbb{R}^{n\times(m-r)}$, and m is the initial value of the hidden dimensionality. The integer r denotes the number of free parameters in $\boldsymbol{\Sigma}_y^r$ with $0 \leqslant r \leqslant m$. We denote this type of parameterizations as FA-r, where the noise covariance is the same as FA-a and FA-b. For any $r \in [0, m]$, FA-r is ML-equivalent to FA-a, and r indicates to what extent FA-r is similar to FA-b. Specially, FA-r becomes FA-a when $r = 0$, and becomes FA-b when $r = m$.

4 VB

Bayesian approach has been extensively used in many scientific areas. One important and difficult problem is computing the marginal likelihood of a given training data set, which involves a high dimensional integral over all parameters. Developed recently, VB [20,21] tackles the integral by means of variational methods to approximate the log marginal likelihood $\ln q(X_N|m,\Xi)$ of a given data set X_N with a lower bound:

$$\begin{aligned}&\mathcal{F}(p(\boldsymbol{\Theta}), p(Y), m, \Xi)\\ &= \int p(\boldsymbol{\Theta})p(Y)\ln\frac{q(X_N, Y|\boldsymbol{\Theta})q(\boldsymbol{\Theta}|m,\Xi)}{p(\boldsymbol{\Theta})p(Y)}\,\mathrm{d}\boldsymbol{\Theta}\,\mathrm{d}Y \quad (7)\\ &= \ln q(X_N|m,\Xi)\\ &\quad -\mathrm{KL}(p(\boldsymbol{\Theta})p(Y)\|q(\boldsymbol{\Theta}, Y|X_N, m, \Xi)), \quad (8)\end{aligned}$$

where Y represents hidden variables, $q(\boldsymbol{\Theta}|m,\Xi)$ is a given prior over the parameters $\boldsymbol{\Theta}$, and $\mathrm{KL}(p\|q) = \int p\ln(p/q) \geqslant 0$ is the KL-divergence, $q(\boldsymbol{\Theta}, Y|X_N, m, \Xi) \propto q(X_N, Y|\boldsymbol{\Theta})q(\boldsymbol{\Theta}|m,\Xi)$. The lower bound $\mathcal{F}$ is a functional of model scale m, prior's hyperparameters Ξ, and the variational posterior $p(\boldsymbol{\Theta})p(Y)$, which is usually assumed to be further factorized as $\prod_i p(\theta_i)\prod_t p(\boldsymbol{y}_t)$ with $\boldsymbol{\Theta} = \{\theta_i\}$ and $Y = \{\boldsymbol{y}_t\}$, in order to obtain computable variational posteriors. The tightness of the bound depends on the KL divergence between the computed variational posterior and the exact Bayesian posterior. By Eq. (7), the model scale is estimated through a two-stage procedure given in Table 2. The optimized $\mathcal{F}$ approaches the maximum log marginal likelihood, which encodes a preference for simpler, more constrained models through assigning higher probability to the data set.

Table 2 Two-stage procedure of VB learning (The two-stage procedure of VB learning for a model selection problem consists of repeating a VBEM algorithm to maximize $\mathcal{F}$ and a discrete maximization to select an appropriate model scale, where τ is the iteration indicator, and τ_o denotes the number of iterations used to reach convergence (i.e., the objective function values vary small). A general derivation of VBEM is referred to the Theorem 2.1 in Ref. [21].)

Stage I: Enumerate each candidate model scale $m \in \mathcal{M}$:
(a.1) $p^{(\tau+1)}(Y) = \arg\max_{p(Y)} \mathcal{F}(p^{(\tau)}(\boldsymbol{\Theta}),\, p(Y), m, \Xi^{(\tau)})$,
(a.2) $p^{(\tau+1)}(\boldsymbol{\Theta}) = \arg\max_{p(\boldsymbol{\Theta})} \mathcal{F}(p(\boldsymbol{\Theta}),\, p^{(\tau+1)}(Y), m, \Xi^{(\tau)})$,
(b) $\Xi^{(\tau+1)} = \arg\max_{\Xi} \mathcal{F}(p^{(\tau+1)}(\boldsymbol{\Theta}), p^{(\tau+1)}(Y),\, m, \Xi)$.
Stage II: Model selection:
$\hat{m} = \arg\max_{m\in\mathcal{M}} \mathcal{F}(p^{(\tau_o)}(\boldsymbol{\Theta}),\, p^{(\tau_o)}(Y), m, \Xi^{(\tau_o)})$.

There have been efforts of VB learning on FA-a [8–10], in which the adopted priors on FA-a's parameters are listed in the left column of Table 3. For FA-b, we have derived a VB learning algorithm in Ref. [26] by the priors given in the right column of Table 3. We directly use the existing VB learning algorithms for FA-a and then extend it for FA-b by certain modifications, and further extend them into a VB learning algorithm for FA-r for which the detailed algorithm is given in Appendix A.

Table 3 Priors distributions of FA-a and FA-b (The above prior distributions in the left column for FA-a have been used in Refs. [8–10]. The priors in the right column for FA-b have been used in Ref. [26]. $\Gamma(z|a,b) = b^a z^{a-1}\mathrm{e}^{-bz}/\Gamma(a)$ is the Gamma density with shape parameter a and inverse scale parameter b, where $\Gamma(a)$ is the Gamma function. The Ξ_a and Ξ_b denote the hyperparameters.)

priors for FA-a	priors for FA-b
$\Xi_a = \{\boldsymbol{a}^\alpha, \boldsymbol{b}^\alpha, a^\varphi, b^\varphi\}$	$\Xi_b = \{\boldsymbol{a}^\nu, \boldsymbol{b}^\nu, a^\varphi, b^\varphi\}$
$\boldsymbol{\varphi} = \varphi\boldsymbol{I}_n = \boldsymbol{\Sigma}_e^{-1}$	$\boldsymbol{\varphi} = \varphi\boldsymbol{I}_n = \boldsymbol{\Sigma}_e^{-1}$, $\boldsymbol{\nu} = \boldsymbol{\Lambda}^{-1}$
$\boldsymbol{a}_i$: ith column vector of $\boldsymbol{A}$	$\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = \boldsymbol{I}_m$, $\boldsymbol{U}$ is at Stiefel manifold
$q(\boldsymbol{A}|\boldsymbol{\alpha}) = \prod_{i=1}^m G(\boldsymbol{a}_i|\boldsymbol{0}, \frac{1}{\alpha_i}\boldsymbol{I}_n)$	$q(\boldsymbol{U}) = 2^{-m}\prod_i \Gamma((n-i+1)/2)\pi^{-(n-i+1)/2}$
$q(\boldsymbol{\alpha}|\boldsymbol{a}^\alpha, \boldsymbol{b}^\alpha) = \prod_{i=1}^m \Gamma(\alpha_i|a_i^\alpha, b_i^\alpha)$	$q(\boldsymbol{\nu}|a^\nu, b^\nu) = \prod_i \Gamma(\nu_i|a_i^\nu, b_i^\nu)$
$q(\varphi|a^\varphi, b^\varphi) = \Gamma(\varphi|a^\varphi, b^\varphi)$	$q(\varphi|a^\varphi, b^\varphi) = \Gamma(\varphi|a^\varphi, b^\varphi)$

The algorithm aims at maximizing the following $\mathcal{F}$ resulted from putting the details of Eq. (6) and Tables 1 and 3 into the variational lower bound $\mathcal{F}$ by Eq. (7):

$$\mathcal{F} = \int\left\{\mathcal{F}_1 + \ln\left[\left(\prod_{i=1}^r \Gamma(\nu_i|a_i^\nu, b_i^\nu)\prod_{k=1}^{m-r}(G(\boldsymbol{a}_k|\boldsymbol{0}, \alpha_k^{-1}\boldsymbol{I}_n)\Gamma(\alpha_k|a_k^\alpha, b_k^\alpha))\right)\frac{q(\boldsymbol{U}_r)\Gamma(\varphi|a^\varphi, b^\varphi)}{p_A p_U p_\nu p_\alpha p_\varphi}\right]\right\} p_{\boldsymbol{\Theta}} p_Y \mathrm{d}\boldsymbol{\Theta}\mathrm{d}Y, \quad (9)$$

$$\mathcal{F}_1 = \sum_{t=1}^N\{\ln G(\boldsymbol{x}_t|\boldsymbol{V}_r\boldsymbol{y}_t, \varphi^{-1}\boldsymbol{I}_n) + \ln G(\boldsymbol{y}_t|\boldsymbol{0}, \mathrm{diag}[\nu_r^{-1}, \boldsymbol{I}_{m-r}]) - \ln p(Y)\}p_Y\mathrm{d}Y, \quad (10)$$

where the variational posterior $p_Y = p(Y)$, $p_{\boldsymbol{\Theta}} = p(\boldsymbol{\Theta}) = p_A p_U p_\alpha p_\nu p_\varphi$, $\boldsymbol{V}_r = [\boldsymbol{U}_r, \boldsymbol{A}_{m-r}]$, $q(\boldsymbol{U}_r) = 2^{-r}\prod_i \Gamma((n-i+1)/2)\pi^{-(n-i+1)/2}$, $\boldsymbol{A}_{m-r} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_{m-r}]$, $\boldsymbol{\nu}_r = [\nu_1, \nu_2, \ldots, \nu_r]$. For simplicity, we omit the subscripts r and $m-r$ in the rest of context.

Moreover, $\mathcal{F}$ by Eq. (9) consists of a part that is a

function of Ξ_r, that is,

$$\mathcal{F} = \mathcal{F}_h(\Xi_r) + \text{others}. \tag{11}$$

As listed as Stage I(b) in Table 2, we also maximize $\mathcal{F}$ with respect to the hyperparameters $\Xi_r = \{a_k^\alpha, b_k^\alpha, a_i^\nu, b_i^\nu, a^\varphi, b^\varphi\}$, which is implemented in the detailed algorithm given by Appendix A with the help of the gradient of $\mathcal{F}_h(\Xi_r)$ with respect to the hyperparameters Ξ_r. It follows from Eq. (8) that such an update of Ξ_r not only minimizes the KL term leading the variational lower bound $\mathcal{F}$ to approach to $\ln q(X_N|m,\Xi_r)$ but also further maximizes $\ln q(X_N|m,\Xi_r)$.

Leaving the computational details of the VB algorithm for FA-r in Table A1 (see Appendix A), we outline the major updates in Table 4 together with the following remarks:

1) When $r = 0$, the VB algorithm on FA-r equivalently implement the one on FA-a [8], where the variational posteriors p_U and p_ν disappear because $\boldsymbol{U}$ and $\boldsymbol{\nu}$ is empty for $r = 0$.

2) When $r = m$, the VB algorithm on FA-r becomes the one on FA-b [26], where p_U and p_ν take over p_A and $p_{\boldsymbol{\alpha}}$ with $\boldsymbol{U}$ and $\boldsymbol{\nu}$ taking the place of $\boldsymbol{A}$. It is empirically observed that p_φ has different impacts on model selection in FA-a and FA-b as shown by experiments later, although the corresponding two variational posteriors which have similar forms are computed from the same Gamma prior.

3) When $0 < r < m$, the VB algorithms on FA-r are variants in addition to those on FA-a and FA-b. On one hand, if we consider no priors over all the parameters $\boldsymbol{\Theta}_m^r$ in FA-r, then the bound $\mathcal{F}$ degenerates to $\mathcal{F}_1$ by Eq. (10). Maximizing $\mathcal{F}_1$ leads to an EM algorithm for FA-r. On the other hand, maximizing $\mathcal{F}_1$ takes the lead in maximizing $\mathcal{F}$ (for a large r), especially when the sample size N or the dimensionality n is very large, because we use a point estimation for p_U (see a.2 in Table 4) and thus the contribution of updating $\boldsymbol{U}$ at Stage I of Table 2 to maximizing $\mathcal{F}$ actually comes through maximizing $\mathcal{F}_1$. Denote the number of free parameters in ϕ by $d(\phi)$, we have $d(\boldsymbol{U}) = nr - 0.5r(r+1)$ and $d(\boldsymbol{\Theta}_m^r) = nm - 0.5r(r-1) + 1$. It follows $d(\boldsymbol{U})/d(\boldsymbol{\Theta}_m^r) \approx r/m$ for a large n and $r/m \approx 1$ for a r close to m that a large n implies the learning on $\boldsymbol{U}$ actually plays a main role in maximizing $\mathcal{F}$. This degeneracy would make the VB algorithm of maximizing $\mathcal{F}$ return back towards the EM algorithm, deteriorating the model selection performances and also reducing the performance differences of FA-r for different r. Still, maximizing $\mathcal{F}$ (for $r > 0$) yields better performance than the algorithm in Refs. [8,10] for FA-a as will be shown later. Moreover, a further improvement in model selection by $\mathcal{F}$ is possible by finding a better prior on $\boldsymbol{U}$.

Table 4 An outline of VB algorithm on FA-r, with details in Table A1 of Appendix A

	the τth iteration of Stage I(a):
(a.1):	Update $p_Y^{(\tau)} = \prod_{t=1}^N G(\boldsymbol{y}_t|\boldsymbol{\mu}_{y|x}^{(t)}, \boldsymbol{\Sigma}_{y|x})$ based on $p_A^{(\tau-1)}$, $p_{\boldsymbol{\alpha}}^{(\tau-1)}$, $p_\nu^{(\tau-1)}$, $p_\varphi^{(\tau-1)}$, $\Xi_r^{(\tau-1)}$.
(a.2):	
	• Update $p_A^{(\tau)} = \prod_{j=1}^n G(\boldsymbol{a}_j|\boldsymbol{\mu}_{A,j}, \boldsymbol{\Sigma}_{A,j})$ based on $p_Y^{(\tau)}$, $p_{\boldsymbol{\alpha}}^{(\tau-1)}$, $p_\nu^{(\tau-1)}$, $p_\varphi^{(\tau-1)}$, $\Xi_r^{(\tau-1)}$.
	• Update $p_U^{(\tau)} \approx \delta(\boldsymbol{U} - \boldsymbol{U}_S^*)$, $\boldsymbol{U}_S^* = \boldsymbol{U}_E^* \left(\boldsymbol{U}_E^{*\mathrm{T}} \boldsymbol{U}_E^*\right)^{-\frac{1}{2}}$, $\boldsymbol{U}_E^* = \left(\sum_t \boldsymbol{x}_t (\boldsymbol{\mu}_{y|x}^{(t)})^{\mathrm{T}}\right) \left(\mathrm{E}\left[\boldsymbol{y}_t \boldsymbol{y}_t^{\mathrm{T}}\right]\right)^{-1}$.
	• Update $p_{\boldsymbol{\alpha}}^{(\tau)} = \prod_{k=1}^{m-r} \Gamma(\alpha_k|\hat{a}_k^\alpha, \hat{b}_k^\alpha)$, based on $p_A^{(\tau)}$, $p_{\boldsymbol{\alpha}}^{(\tau)}$, $p_\nu^{(\tau-1)}$, $p_\varphi^{(\tau-1)}$, $\Xi_r^{(\tau-1)}$.
	• Update $p_\nu^{(\tau)} = \prod_{i=1}^r \Gamma(\nu_i|\hat{a}_i^\nu, \hat{b}_i^\nu)$, based on $p_A^{(\tau)}$, $p_{\boldsymbol{\alpha}}^{(\tau)}$, $p_\nu^{(\tau)}$, $p_\varphi^{(\tau-1)}$, $\Xi_r^{(\tau-1)}$.
	• Update $p_\varphi^{(\tau)} = \Gamma(\varphi|\hat{a}^\varphi, \hat{b}^\varphi)$, based on $p_A^{(\tau)}$, $p_{\boldsymbol{\alpha}}^{(\tau)}$, $p_\nu^{(\tau)}$, $p_\varphi^{(\tau)}$, $\Xi_r^{(\tau-1)}$.
	the τth iteration of Stage I(b):
	Update hyperparameters Ξ_r by gradient method, $\Xi_r^{\text{new}} = \Xi_r^{\text{old}} + \eta \left.\frac{\partial \mathcal{F}_h(\Xi_r)}{\partial \Xi_r}\right|_{\Xi_r = \Xi_r^{\text{old}}}$.

5 BYY harmony learning

Firstly proposed in Ref. [27] and systematically developed over a decade, BYY harmony learning theory is a general statistical learning framework that can handle both parameter learning and model selection under a best harmony principle. The BYY harmony learning on typical structures leads to new model selection criteria, new techniques for implementing regularization and a class of algorithms that implement automatic model selection during parameter learning. In the sequel, we introduce some fundamentals of BYY. Readers are referred to Ref. [22] for a recent systematic introduction.

Mathematically, the best harmony principle is to maximize the following harmony functional:

$$H(p\|q) = \int p(R|X)p(X) \ln\left[q(X|R)q(R)\right] \mathrm{d}X \mathrm{d}R = \int p(\boldsymbol{\Theta}|X) H(p\|q,\boldsymbol{\Theta}) \mathrm{d}\boldsymbol{\Theta},$$
$$H(p\|q,\boldsymbol{\Theta}) = \int p(Y|X,\boldsymbol{\Theta})p(X) \ln[q(X|Y,\boldsymbol{\Theta})q(Y|\boldsymbol{\Theta})] \mathrm{d}Y \mathrm{d}X + \ln q(\boldsymbol{\Theta}|\Xi), \tag{12}$$

where the observation data X are generated from its inner representation $R = \{Y, \boldsymbol{\Theta}\}$, where a parameter set $\boldsymbol{\Theta}$ represents the underlying regularities of X and Y is the inner representation of X accordingly. The two types of Bayesian decompositions, i.e., $p(R|X)p(X)$ and $q(X|R)q(R)$, are called Yang machine and Ying machine

respectively, which form a BYY system as depicted in Fig. 1.

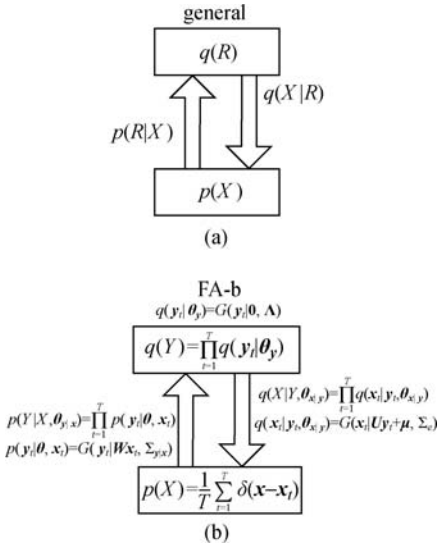


Fig. 1 BYY system in the general form and specific structures for FA

An important nature of maximizing $H(p\|q)$ is that it leads to not only a best matching between the Ying-Yang pair, but also a compact model with a least complexity. Such an ability can be observed and investigated from several perspectives, see Sect. 4.1 in Ref. [22], and here we only introduce one of them due to space limit. On one hand, maximizing $H(p\|q)$ forces Ying machine $q(X|R)q(R)$ to match Yang machine $p(R|X)p(X)$. Due to a finite sample size and practical constraints imposed on the Ying-Yang structures, a perfect equality $q(X|R)q(R) = p(R|X)p(X)$ may not be really reached but still be approached as possible as it can. At this equality, $H(p\|q)$ becomes the negative entropy that describes the complexity of the system. Further maximizing it will decrease the system complexity which provides a model selection ability.

In implementation, we maximize $H(p\|q)$ by a two-stage procedure as shown in Table 5, which shares a format similar to the one in Table 2 and also the conventional two-stage procedure introduced after Eqs. (3) and (4). Moreover, the BYY harmony learning is also featured by its favorable nature that model selection is made automatically during the implementation of merely Stage I, e.g., for FA-b in Table 1, the implementation of either Stage I(a) or both Stage I(a) and I(b) will drive some λ_j to zero when the jth dimension of $\boldsymbol{y}_t$ is extra. Thus, automatic model selection can be made via discarding the jth dimension after checking $\lambda_j \to 0$.

This paper mainly focuses on a detailed comparative study with the VB learning in Table 2 by the conventional two-stage procedure, without making automatic model selection via checking $\lambda_j \to 0$. Also, we provide a simple comparative investigation on the automatic model selection performances of BYY and VB. Further details about automatic model selection are referred to Sects. 2.1 and 3.2 in Ref. [22] and to Sect. 2.2 in Ref. [28] for further improvements via exploring a co-dimensional matrix pair nature (additionally where an improved model selection criterion is given by e.g., Eq. (29) in Ref. [28]).

Table 5 General two-stage iterative BYY harmony learning procedure (The procedure is restated from Fig. 6(a) in Ref. [29] (also see Eqs. (6) and (7) in Ref. [30] and Fig. 5(b) in Ref. [22]), where $n_f(\boldsymbol{\Theta})$ is the number of free parameters in $\boldsymbol{\Theta}$, and $d_m(\Xi)$ is given in Eq. (16). The "incr" means "*to increase*".)

Stage I:	Enumerate candidate models by m and for each candidate, we iterate the following (a) and (b) until converged: (a) $\boldsymbol{\Theta}^{(\tau)} = \arg\max / \mathrm{incr}_{\boldsymbol{\Theta}} H(p\|q, \boldsymbol{\Theta}, m, \Xi^{(\tau-1)})$, (b) $\Xi^{(\tau)} = \arg\max / \mathrm{incr}_{\Xi} \Big\{ H(p\|q, \boldsymbol{\Theta}^{(\tau)}, m, \Xi) + \frac{1}{2} d_m(\Xi) + H_b(m, \Xi) \Big\}$,
Stage II:	Select the best $\hat{m}$: $\hat{m} = \arg\min_m \Big\{ - H(p\|q, \boldsymbol{\Theta}^{(\tau^*)}, m, \Xi^{(\tau^*)}) + \frac{1}{2} n_f(\boldsymbol{\Theta}_m) - H_b(m, \Xi) \Big\}$, τ^* is the value of the iteration indicator τ when Stage I converged.

Next, we outline the derivation of Table 5 with details referred to Sect. 4.3 in Ref. [22]. Putting the empirical density $p(X) = \delta(X - X_N)$ with $X_N = \{\boldsymbol{x}_t\}_{t=1}^N$ into Eq. (12) and splitting $\boldsymbol{\Theta} = \boldsymbol{\Theta}^a \cup \boldsymbol{\Theta}^b$, $\boldsymbol{\Theta}^a \cap \boldsymbol{\Theta}^b =$ empty, we have

$$H(p\|q) = H_b(m, \Xi) + \int p(\boldsymbol{\Theta}|X_N, \Xi) H(p\|q, \boldsymbol{\Theta}, m, \Xi)\, \mathrm{d}\boldsymbol{\Theta}, \tag{13}$$

$$H_b(m, \Xi) = \int p(\boldsymbol{\Theta}^b|X_N, \Xi) \ln q(\boldsymbol{\Theta}^b|\Xi)\, \mathrm{d}\boldsymbol{\Theta}^b, \tag{14}$$

$$H(p\|q, \boldsymbol{\Theta}, m, \Xi) = \int p(Y|X_N, \boldsymbol{\Theta}) \ln [q(X_N|Y, \boldsymbol{\Theta}) q(Y|\boldsymbol{\Theta})]\, \mathrm{d}Y + \ln q(\boldsymbol{\Theta}^a|\Xi), \tag{15}$$

$$\int p(\boldsymbol{\Theta}|X_N, \Xi) H(p\|q, \boldsymbol{\Theta}, m, \Xi)\, \mathrm{d}\boldsymbol{\Theta} \approx H(p\|q, \boldsymbol{\Theta}^*, m, \Xi) + \frac{1}{2} d_m(\Xi), \tag{16}$$

$$d_m(\Xi) = -n_f(\boldsymbol{\Theta}) + (\boldsymbol{\Theta}^X - \boldsymbol{\Theta}^*)^{\mathrm{T}} \Omega(\boldsymbol{\Theta}^*, \Xi)(\boldsymbol{\Theta}^X - \boldsymbol{\Theta}^*),$$
$$\boldsymbol{\Theta}^* = \arg\max_{\boldsymbol{\Theta}} H(p\|q, \boldsymbol{\Theta}, m, \Xi), \tag{17}$$

where the integral for $H_b(m, \Xi)$ can be solved analytically and $\boldsymbol{\Theta}^b$ could be an empty subset. The second term in Eq. (13) is handled by the so called apex approximation, resulting in Eq. (16), where $\Omega(\boldsymbol{\Theta}^*, \Xi) = \nabla^2_{\boldsymbol{\Theta}\boldsymbol{\Theta}^{\mathrm{T}}} H(p\|q, \boldsymbol{\Theta}, m, \Xi)$ is the Hessian matrix evaluated at $\boldsymbol{\Theta}^*$. $\boldsymbol{\Theta}^X$ is the mean of $p(\boldsymbol{\Theta}|X_N, \Xi)$. Simply, we adopt $\boldsymbol{\Theta}^X = \boldsymbol{\Theta}^{(\tau-1)}$ and thus $\boldsymbol{\Theta}^* = \boldsymbol{\Theta}^{(\tau)}$. It follows that $\boldsymbol{\Theta}^{(\tau)} - \boldsymbol{\Theta}^{(\tau-1)}$ vanishes when the iteration converges.

Therefore, we get Stage I(a) in Table 5 directly from Eq. (17). Moreover, putting Eq. (16) into Eq. (13), we may update the hyperparameters Ξ by Stage I(b) and select $\hat{m}$ by Stage II.

Specifically, we consider the FA-r model by Eq. (6) with i.i.d. samples in $X_N = \{\boldsymbol{x}_t\}_{t=1}^N$, from which we have

$$\begin{aligned}
&q(X|Y,\boldsymbol{\Theta}) = \prod_t q(\boldsymbol{x}_t|\boldsymbol{y}_t,\boldsymbol{\Theta}),\\
&q(Y|\boldsymbol{\Theta}) = \prod_t q(\boldsymbol{y}_t|\boldsymbol{\Theta}),\\
&p(Y|X,\boldsymbol{\Theta}) = \prod_t p(\boldsymbol{y}_t|\boldsymbol{x}_t,\boldsymbol{\Theta}),\\
&q(\boldsymbol{x}|\boldsymbol{y},\boldsymbol{\Theta}) = G(\boldsymbol{x}|\boldsymbol{V}_r\boldsymbol{y}+\boldsymbol{\mu},\boldsymbol{\Sigma}_e),\\
&q(\boldsymbol{y}|\boldsymbol{\Theta}) = G(\boldsymbol{y}|\boldsymbol{0},\boldsymbol{\Sigma}_y^r),\\
&p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{\Theta}) = G(\boldsymbol{y}|\widetilde{\boldsymbol{W}}\boldsymbol{x}+\boldsymbol{w},\boldsymbol{\Sigma}_{y|x}),
\end{aligned} \tag{18}$$

where the Yang machine $G(\boldsymbol{y}|\widetilde{\boldsymbol{W}}\boldsymbol{x},\boldsymbol{\Sigma}_{y|x})$ is designed as the inverse from $G(\boldsymbol{x}|\boldsymbol{V}_r\boldsymbol{y}+\boldsymbol{\mu},\boldsymbol{\Sigma}_e)$ and $q(\boldsymbol{y}|\boldsymbol{\Theta}) = G(\boldsymbol{y}|\boldsymbol{0},\boldsymbol{\Sigma}_y^r)$ according to the variety preservation (VP) principle (see Eq. (31) in Ref. [22]). Moreover, it follows from $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{x}+\boldsymbol{w}$ within Eq. (47) in Ref. [30] that we consider free parameters $\widetilde{\boldsymbol{W}}$ and $\boldsymbol{w}$ to be updated via learning.

In the sequel, we develop the learning procedure in Table 5 into a gradient based BYY learning algorithm on FA-r. Leaving the computational details of this algorithm in Table B1 (see Appendix B), here we introduce its key points in an outline in Table 6.

At the τth step of the implementation, putting Eq. (16) into Eq. (13) and obtaining $\boldsymbol{\Theta}_r^{(\tau)}$ of the FA-r model by Eq. (6), we have

$$H(p\|q) \approx H(p\|q,\boldsymbol{\Theta},m,\Xi) + \frac{1}{2}d_m(\Xi_r) + H_b, \tag{19}$$

$$\begin{aligned}
&d_m(\Xi_r) = -n_f(\boldsymbol{\Theta}_r) + \boldsymbol{\Delta}_{\boldsymbol{\Theta}_r}^{\mathrm{T}}\,\Omega(\boldsymbol{\Theta}_r^{(\tau)},\Xi_r)\,\boldsymbol{\Delta}_{\boldsymbol{\Theta}_r},\\
&\boldsymbol{\Delta}_{\boldsymbol{\Theta}_r} = \boldsymbol{\Theta}_r^{(\tau-1)} - \boldsymbol{\Theta}_r^{(\tau)},
\end{aligned} \tag{20}$$

from which we get Stage I(b) in Table 6 for updating the hyperparameters Ξ at the τth step.

Further putting Eq. (18) and the priors given in Table 3 into Eqs. (13) and (15), we have

$$\begin{aligned}
&H(p\|q,\boldsymbol{\Theta},m,\Xi)\\
&\quad = \prod_t \ln G(\boldsymbol{x}_t|\boldsymbol{0},\boldsymbol{\Sigma}_x) - N\ln\sqrt{(2\pi\mathrm{e})^m|\boldsymbol{\Sigma}_{y|x}|}\\
&\qquad + d_r(\widetilde{\boldsymbol{W}}) + \ln q(\boldsymbol{\Theta}^a|\Xi),
\end{aligned} \tag{21}$$

$$\begin{cases}
\boldsymbol{\Sigma}_x = \boldsymbol{V}\boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}} + \varphi^{-1}\boldsymbol{I}_n,\\
\boldsymbol{\Sigma}_y^r = \mathrm{diag}[\boldsymbol{\nu}^{-1},\boldsymbol{I}_{m-r}],\\
\boldsymbol{\Sigma}_{y|x} = \left[(\boldsymbol{\Sigma}_y^r)^{-1} + \boldsymbol{V}^{\mathrm{T}}(\varphi\boldsymbol{I}_n)\boldsymbol{V}\right]^{-1},
\end{cases} \tag{22}$$

$$\begin{cases}
d_r(\widetilde{\boldsymbol{W}}) = -\frac{1}{2}\mathrm{Tr}(\boldsymbol{\Delta}_W^{\mathrm{T}}\boldsymbol{\Sigma}_{y|x}^{-1}\boldsymbol{\Delta}_W\boldsymbol{S}_N),\\
\boldsymbol{S}_N = \sum_t \boldsymbol{x}_t\boldsymbol{x}_t^{\mathrm{T}},\\
\boldsymbol{\Delta}_W = \widetilde{\boldsymbol{W}} - \boldsymbol{W};\ \boldsymbol{W} = \boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}\boldsymbol{\Sigma}_x^{-1}.
\end{cases} \tag{23}$$

Table 6 A sketch of the gradient implementation of BYY learning algorithm on FA-r (All computational details are referred to Table B1 in Appendix B.)

objective: maximize the harmony functional $H(p\|q) \approx H_1 + d_r(\widetilde{\boldsymbol{W}}) + \ln q(\boldsymbol{\Theta}^a|\Xi) + H_b + \frac{1}{2}d_m(\Xi)$, $H_1 = -\frac{N(n+m)}{2}\ln(2\pi) - \frac{Nm}{2} + \frac{N}{2}\ln|\boldsymbol{\nu}_r| + \frac{N}{2}\ln|\varphi\boldsymbol{I}_n|$ $\quad -\frac{1}{2}\mathrm{Tr}[S_N(\boldsymbol{V}\cdot\mathrm{diag}[\boldsymbol{\nu}^{-1},\boldsymbol{I}_{m-r}]\boldsymbol{V}^{\mathrm{T}} + \varphi^{-1}\boldsymbol{I}_n)^{-1}]$. The last four terms of $H(p\|q)$ are given by Eqs. (23), (24), (25), and (19).
the τth iteration of Stage I(a): gradient method to update the parameters $\boldsymbol{\theta}^{(\tau)} = \boldsymbol{\theta}^{(\tau-1)} + \eta\partial\boldsymbol{\theta}$, $\partial\boldsymbol{\theta} = \partial^H\boldsymbol{\theta} = \frac{\partial H(p\|q)}{\partial\boldsymbol{\theta}}\big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{(\tau-1)}}$, $\forall\boldsymbol{\theta}\in\{\boldsymbol{U},\boldsymbol{A},\boldsymbol{\nu},\varphi\}$, η is a step size. According to the five terms of $H(p\|q)$, we have $\partial\boldsymbol{\theta} = \partial^{H_1}\boldsymbol{\theta} + \partial^{d_r}\boldsymbol{\theta} + \partial^q\boldsymbol{\theta} + \partial^{H_b}\boldsymbol{\theta} + \partial^{d_m}\boldsymbol{\theta}$.
the τth iteration of Stage I(b): gradient method to update the hyperparameters Hessian matrix $\Omega(\boldsymbol{\Theta}^{(\tau)},\Xi)$ (approximated as block-diagonal); $\xi^{(\tau)} = \xi^{(\tau-1)} + \eta\,\frac{\partial H(p\|q)}{\partial\xi}\Big|_{\xi^{(\tau-1)},\boldsymbol{\theta}^{(\tau)}}$, $\forall\xi\in\{a_k^\alpha,b_k^\alpha,a_i^\nu,b_i^\nu,a^\varphi\}$, η is a step-size.

Again, the above $H(p\|q,\boldsymbol{\Theta},m,\Xi)$ shares a format similar to Eq. (4). The term $d_r(\widetilde{\boldsymbol{W}})$ vanishes when the algorithm converges, taking a regularization role during learning for alleviating to be stuck at local optimums. The previous studies of the BYY learning for FA-a in Ref. [18] or for FA-b in Ref. [19] without considering the prior term $\ln q(\boldsymbol{\Theta}^a|\Xi)$, except a preliminary study made in Ref. [26]. In contrast, a role similar to the conventional Bayesian regularization is taken by the (log) prior term in Eq. (21) with the following details:

$$\begin{aligned}
&\ln q(\boldsymbol{\Theta}^a|\Xi)\\
&\quad = \ln\left[q(\boldsymbol{U})\prod_{i=1}^r\Gamma(\nu_i|a_i^\nu,b_i^\nu)\Gamma(\varphi|a^\varphi,b^\varphi)\right]\\
&\quad = -r\ln 2 + \sum_{i=1}^r\left[\ln\Gamma\left(\frac{n-i+1}{2}\right) - \frac{n-i+1}{2}\ln\pi\right]\\
&\qquad + \sum_{i=1}^r\left\{(a_i^\nu-1)\ln\nu_i - b_i^\nu\nu_i + a_i^\nu\ln b_i^\nu - \ln\Gamma(a_i^\nu)\right\}\\
&\qquad + (a^\varphi-1)\ln\varphi - b^\varphi\ln\varphi + a^\varphi\ln b^\varphi - \ln\Gamma(a^\varphi),
\end{aligned} \tag{24}$$

$$\begin{aligned}
&H_b(m,\Xi)\\
&\quad = \int p(\boldsymbol{\alpha}|\boldsymbol{A},\varphi,X_N)\ln[q(\boldsymbol{A}|\boldsymbol{\alpha})q(\boldsymbol{\alpha})]\mathrm{d}\boldsymbol{\alpha}\\
&\quad = \sum_{k=1}^{m-r}\left\{(\hat{a}_k^\alpha-1)\left(\psi(\hat{a}_k^\alpha) - \ln\hat{b}_k^\alpha\right) - \hat{a}_k^\alpha + a_k^\alpha\ln b_k^\alpha\right.\\
&\qquad \left. - \ln\Gamma(a_k^\alpha)\right\} - \frac{n(m-r)}{2}\ln(2\pi),
\end{aligned} \tag{25}$$

where $p(\boldsymbol{\alpha}|\boldsymbol{A},\varphi,X_N) = \prod_{k=1}^{m-r}\Gamma(\alpha_k|\hat{a}_k^\alpha,\hat{b}_k^\alpha)$ with $\hat{a}_k^\alpha = a_k^\alpha + \frac{n}{2}$ and $\hat{b}_k^\alpha = b_k^\alpha + \frac{\boldsymbol{a}_k^{\mathrm{T}}\boldsymbol{a}_k}{2}$, and $\boldsymbol{I}_\ell$ denotes an $\ell\times\ell$ identity matrix, and $\psi(\cdot)$ is the digamma function.

Putting the above obtained $H(p\|q,\boldsymbol{\Theta},m,\Xi)$ into Table 6, we can derive the detailed equations for gradients

and Hessian matrices (with respect to each part of unknown parameters), from which we obtained the BYY learning algorithm for FA-r given in Table B1 (see Appendix B) together with the following remarks:

1) When $r = 0$, Table 6 implements BYY harmony learning on FA-a, where the terms $\ln|\boldsymbol{\nu}|$, $\ln q(\boldsymbol{U})$, and $\ln q(\boldsymbol{\nu})$ in $H(p\|q)$ disappear.

2) When $r = m$, Table 6 implements BYY harmony learning on FA-b, where the term H_b given in Eq. (25) disappears, and maximizing the term $\ln|\boldsymbol{\nu}|$ pushes $1/\nu_i \to 0$ if the ith hidden dimension is an extra scale.

3) When $0 < r < m$, Table 6 provides variants of BYY learning algorithms on FA between FA-a and FA-b.

Last but not the least, the algorithm in Table B1 is derived from getting the integral over $\boldsymbol{y}$ analytically removed and then making gradient based updates. Alternatively, maximizing the harmony functional can also be implemented by a Ying-Yang alternation procedure (see e.g., Fig. 8 of Ref. [22]), which is featured by getting the peak value of $\boldsymbol{y}^*$ in the Yang step and removing the integral over $\boldsymbol{y}$ around this $\boldsymbol{y}^*$, while the Ying step updates all the unknown parameters. Readers are referred to Sect. 4.3 in Ref. [22] for more details.

6 Empirical analysis

6.1 Three levels of investigations

This empirical analysis has the following purposes:

1) Examining whether FA-b is better than FA-a for making model selection, via BYY, VB, AIC, BIC, and DNLL;

2) Examining the joint effects of two parameterizations and the role of priors on the performances of model selection;

3) Comparing the performances of BYY, VB, AIC, BIC, and DNLL.

Towards these purposes, we conduct investigations at three different levels, as shown in Table 7. The criteria AIC, BIC, and DNLL are indifferent for FA-b and FA-a in term of making model selection. Without a priori $q(\boldsymbol{\Theta}^a|\Xi)$ (i.e., Level 1 in Table 7), VB degenerates to ML and thus is also indifferent for FA-b and FA-a. In this case, only BYY is capable of model selection, and has different performances on FA-a and FA-b. To enable VB to make model selection, we take a priori $q(\boldsymbol{\Theta}^a|\Xi)$ in consideration to compare the performances of both BYY and VB. Since $q(\boldsymbol{\Theta}^a|\Xi)$ depends on the hyperparameters Ξ, it is natural to consider the cases with Ξ fixed (i.e., Level 2 in Table 7) and the cases with Ξ optimized (i.e., Level 3 in Table 7) via maximizing the variational lower bound $\mathcal{F}$ by VB and $H(p\|q)$ by BYY.

For simplicity and clarity, we use the notations VB(r,l) and BYY(r,l) to indicate the two-stage procedure by VB and BYY, respectively, for different values of r for FA-r and for different levels of l. E.g., VB(r,1), VB(r,2), VB(r,3) versus BYY(r,1), BYY(r,2), BYY(r,3), respectively. Also, on FA-a and FA-b we have VB(a,1), VB(b,1) (i.e., VB(0,1), VB(m,1)) versus BYY(a,1), BYY(b,1) (i.e., BYY(0,1), BYY(m,1)).

We adopt the empirical analysis method presented in Ref. [31] for the performance evaluation on the three levels of implementations of VB and BYY for FA-a and FA-b, and also with the performances on AIC, BIC and DNLL included for comparisons.

The simulated data sets are randomly generated according to FA-b (or FA-a) in Table 1. A setting $\mathcal{S}(N, \gamma_o, n, m^*)$ for FA-b is determined by choosing values from a candidate set of the sample sizes N, the signal-to-noise ratios (SNRs) γ_o, the dimensionality of the observed variable $n = \dim(\boldsymbol{x})$ and the dimensionality of the latent variable $m^* = \dim(\boldsymbol{y})$, where SNR is defined as the ratio of the m^*th largest eigenvalue of the population covariance matrix $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}} + \sigma_e^2\boldsymbol{I}_n$ to the noise variance σ_e^2, i.e., $\gamma_o = (\lambda_{m^*} + \sigma_e^2)/\sigma_e^2$.

Listed in Table 8 are the choices of $\mathcal{S}(N, \gamma_o, n, m^*)$ considered in this paper. For example, $\mathcal{S}(50, 3.0, 15, 5)$ means that training data sets $X_N = \{\boldsymbol{x}_t\}_{t=1}^N$ are randomly generated according to FA-b with $N = 50$, $\gamma_o = 3.0$, $n = 15$ and $m^* = 5$.

6.2 FA-A versus FA-B: Performances of BYY, VB, AIC, BIC, and DNLL

Each of BYY, VB, AIC, BIC, and DNLL is implemented for 10^3 trials on each of the settings $\mathcal{S}(:,:,15,5) = \{\mathcal{S}(N, \gamma_o, 15, 5) \ : \ \forall N \in V(N), \gamma_o \in V(\gamma_o)\}$ with different sample sizes and SNRs chosen from Table 8. The model selection accuracies are reported in Figs. 2 to 4 through the contour maps suggested in Ref. [31] for illustrating the joint effect of N and γ_o on the performance. Readers are referred to Ref. [31] for the characteristics

Table 7 Three levels of investigations

	VB in Table 2	BYY in Table 5
Level 1: $\ln q(\boldsymbol{\Theta}\|\Xi) = 0$	update p_Y and $\boldsymbol{\Theta} = \arg\max_{\Theta} \mathcal{F}_1$	fix $\partial^q\boldsymbol{\theta} = \partial^{H_b}\boldsymbol{\theta} = \partial^{d_m}\boldsymbol{\theta} = 0$
	instead of $\{p_A, p_U, p_\alpha, p_\nu, p_\varphi\}$;	not update $\Xi_r = \{a_k^\alpha, b_k^\alpha, a_i^\nu, b_i^\nu, a^\varphi, b^\varphi\}$
Level 2: $q(\boldsymbol{\Theta}\|\Xi)$ with Ξ fixed	without Stage I(b)	without Stage I(b)
Level 3: $q(\boldsymbol{\Theta}\|\Xi)$ with Ξ optimized	all the steps	all the steps

Table 8 Candidate values of each feature (All possible combinations consist of all settings $\mathcal{S}(N, \gamma_o, n, m^*)$ used in the empirical analysis. We set $\lambda_1 = \lambda_2 = \cdots = \lambda_{m^*} = 1$. For two-phase procedures, we set the candidate set of hidden dimensionalities as $\mathcal{M} = \{1, 2, \ldots, 9\}$ or $\{1, 2, \ldots, 15\}$ for $m^* = 5, 10$ respectively, unless otherwise specified. $V(f)$ is the set of the candidate values of the feature f.)

features f	candidate values
sample size N	$V(N)$: 25, 50, 75, 100, 200, 400, 800
SNR: $\gamma_o = \frac{\lambda_{m^*}}{\sigma_e^2} + 1$	$V(\gamma_o)$: 1.2, 1.5, 2, 2.5, 3, 3.5, 4, 8, 16.
dim: $\{n, m^*\}$	$V(n, m^*)$: $\{15, 5\}$, $\{30, 10\}$

(e.g., a three-region partition phenomenon) of the contour maps for describing model selection accuracies, as well as a systematic comparison of BYY(b,1) with several classical criteria and recently developed model selection methods.

Here we summarize our observations on Figs. 2 to 4 as follows:

1) Shown by Fig. 3, VB performs better on FA-b than on FA-a. VB(a,1) and VB(b,1) actually implement the maximum likelihood principle which is not good[1)] for model selection under a finite sample size. For a relatively small N, FA-b is obviously superior to FA-a under VB. As N goes large, the difference between VB(b,2) and VB(a,2) tends to be not so obvious, because a large N would lead $\mathcal{F}$ (for $r = m$) in Eq. (9) close to $\mathcal{F}_1$ and thus VB(b,2) approaches to VB(b,1). Analogously, VB(a,2) approaches to VB(a,1). This tendency towards maximum likelihood gradually reduces the gain obtained from using FA-b in place of FA-a.

Due to the approximation $p_U^{(\tau)} \approx \delta(\boldsymbol{U} - \boldsymbol{U}_S^*)$ in Table 4, VB(b,3) with optimized hyperparameters becomes even closer to maximum likelihood than VB(b,2) does, while VB(a,3) does not decline to be inferior to VB(a,2) with the help of the variational posterior over the loading matrix $\boldsymbol{A}$ [10,21]. As a result, the gain of using FA-b in place of FA-a becomes lower as we proceed from VB(b,2) to VB(b,3).

Moreover, on FA-a, VB(a,2) is slightly worse than BIC especially for a small sample size, but on FA-b, VB(b,2) greatly outperforms BIC.

2) Shown by Fig. 4, BYY performs better on FA-b than on FA-a even more significantly. Moreover, with FA-a replaced by FA-b, the gain obtained by BYY is obviously higher than that by VB. Unlike VB, BYY differs from maximum likelihood even without priors by Eq. (15), since FA-b's latent coordinate system better encodes the latent variable $\boldsymbol{y}$'s complexity, which is well captured by BYY (see discussions in Sect. 2.2 and Fig. 5 in Ref. [22]). If the ith hidden dimension is extra, maximizing $H(p\|q)$ with the term $\ln|\boldsymbol{\nu}|$ present for FA-b in Table 6 would push its variance $1/\nu_i \to 0$.

Moreover, the model selection performances of BYY are further improved by adopting a prior $q(\boldsymbol{\Theta}^a|\Xi)$, and further improved by optimizing the hyperparameter Ξ. For most cases (especially for a small sample size), BYY(b,3) outperforms VB(b,2) which is the best among all VB implementations, while VB(b,2) has a relative advantage for the case of large N and small SNR.

6.3 FA-r: Performances of VB versus BYY

6.3.1 Priors affect model selection

The above results show that appropriate priors benefit model selection. Next, based on the family FA-r of FA parameterizations, we present a detailed empirical analysis on how much each part of the priors contributes to model selection performance.

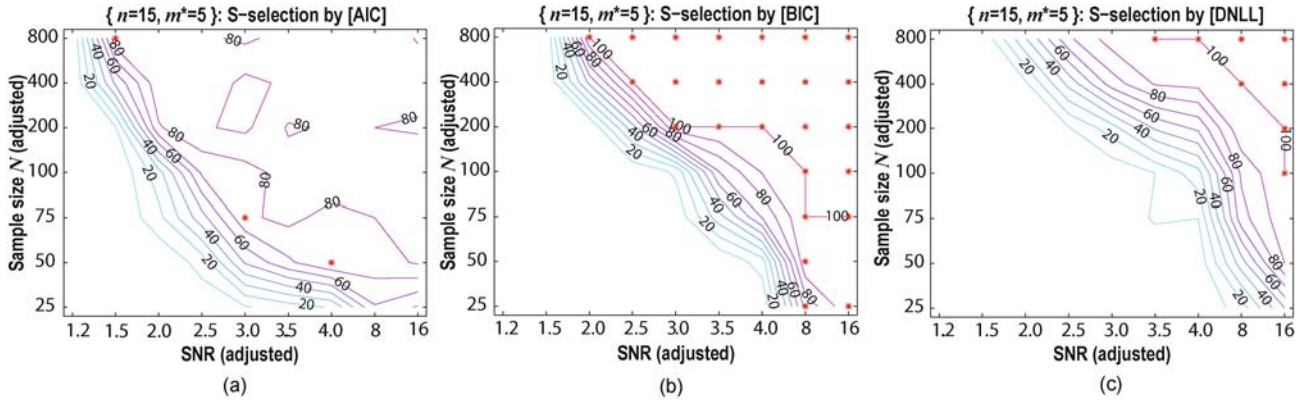


Fig. 2 Successful-selection (S-selection) rates on $\mathcal{S}(:, :, 15, 5)$ presented in terms of contour maps. (The axes are adjusted by equally spacing the elements in $V(N)$ and $V(\gamma_o)$. A red asterisk $(*)$ at the coordinate (N, γ_o) indicates that the corresponding criterion gets the highest successful selection rate on $\mathcal{S}(N, \gamma_o, 15, 5)$ among AIC, BIC, DNLL and all implementations of VB and BYY. Briefly speaking, the closer the contour lines to the bottom-left corner reflects the more robust the corresponding algorithm to small N and γ_o, and the more there are red asterisks, the better the performance. AIC, BIC, and DNLL have the same performance on FA-a and FA-b.)

1) For a finite sample size, the obtained maximum likelihood $L(\hat{\boldsymbol{\Theta}}_m)$ increases as m grows, and thus $\hat{m} = \arg\max_m L(\hat{\boldsymbol{\Theta}}_m)$ tends to overestimation. An alternative criterion is DNLL given by Eq. (5) and Fig. 2(c), which finds the maximum increment in the likelihood function.

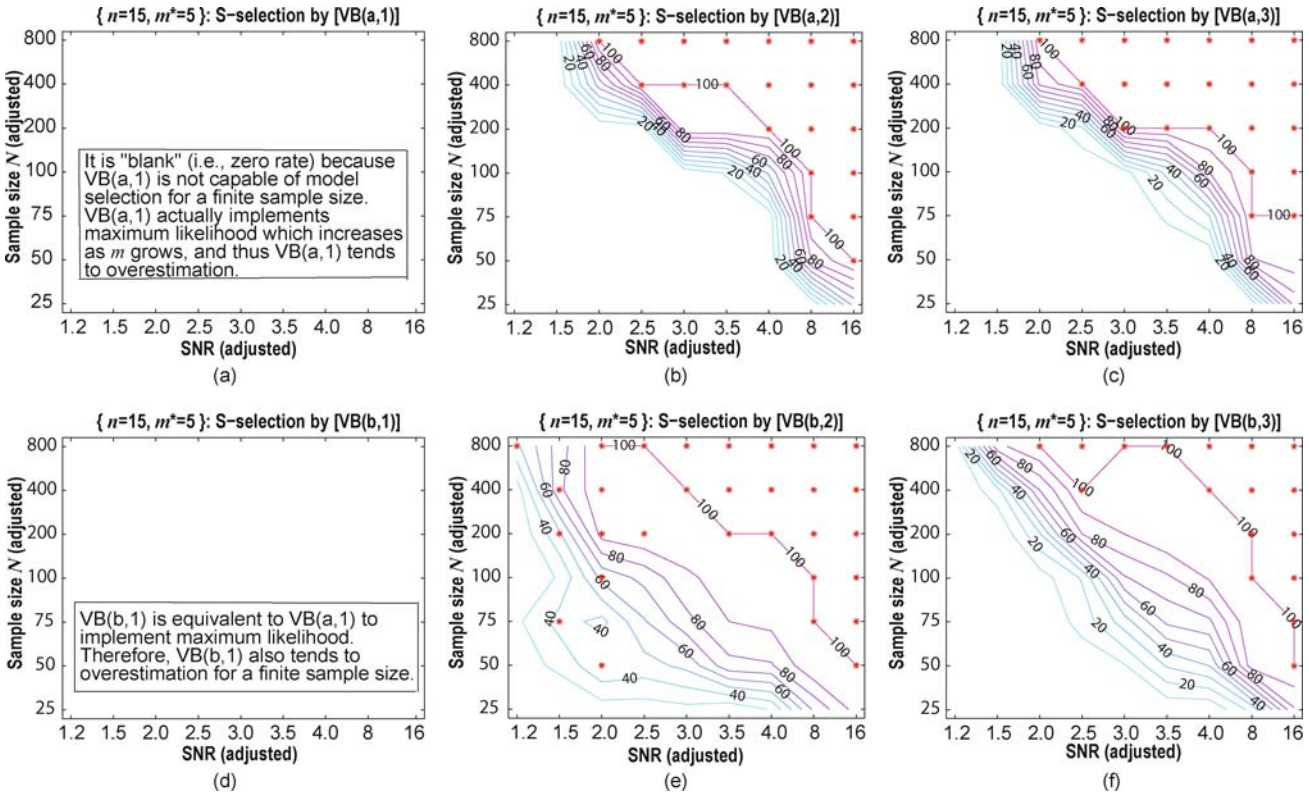


Fig. 3 Successful-selection (S-selection) rates of VB obtained on the same synthetic data as in Fig. 2. (The red asterisk (∗) indicates the corresponding criterion gets the highest model selection accuracy among all VB/BYY implementations as well as AIC, BIC, and DNLL. Notice that the figures of VB(a,1) and VB(b,1) are "blank" (i.e., zero rates of successful-selections). VB(a,1) and VB(b,1) are not capable of model selection for a finite sample size, because they both implement maximum likelihood $L(\hat{\Theta}_m)$ which increases as m grows. Therefore, the estimated $\hat{m} = \arg\max_{m\in\mathcal{M}} L(\hat{\Theta}_m)$ tends to overestimation. In the figures of VB(a,2) and VB(a,3), the accuracies are zero when SNR < 1.5 and $N \leqslant 800$. Actually, as N further goes large, the rates will increase.)

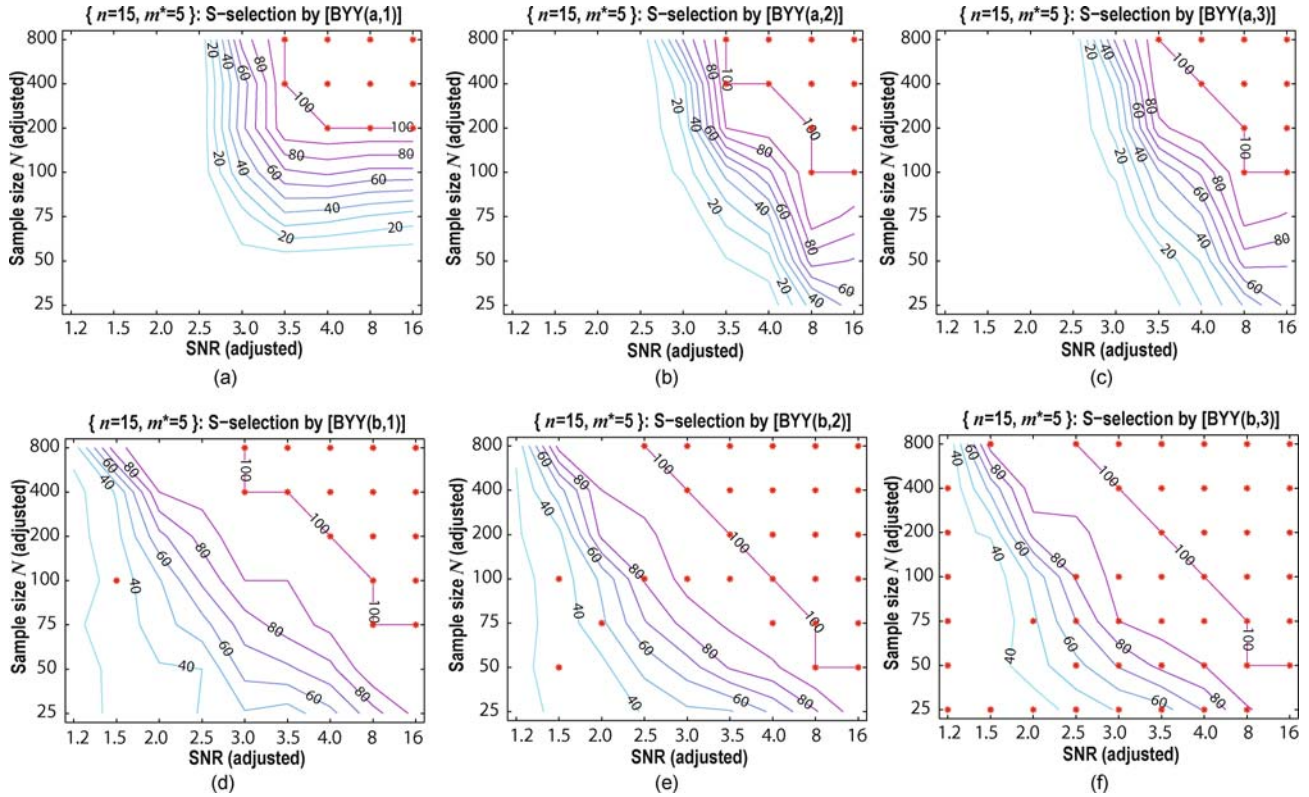


Fig. 4 Successful-selection (S-selection) rates of BYY obtained on the same synthetic data as in Fig. 2. (The red asterisk (∗) indicates the corresponding criterion gets the highest model selection accuracy among all VB/BYY implementations as well as AIC, BIC, and DNLL.)

We use the same configurations $\mathcal{S}(:,:,15,5)$ given in Sect. 6.2. Model selection accuracies are selectively reported in Figs. 5 and 6 for a series of critical settings (N, γ_o), illustrating the relative strengths and weaknesses of different learning algorithms.

Figures 5 and 6 show that the performances are continually improved as parts of priors are incrementally incorporated. The best performance is mostly achieved on the parameterization FA-m^* with $m^* = 5$ being the true number of hidden factors, and the performance of FA-r drops sharply as r reduces from m^* towards FA-a, while the performance of FA-r declines a little and reduces slowly as r grows from m^* towards FA-b. All chosen combinations of priors work better on those of

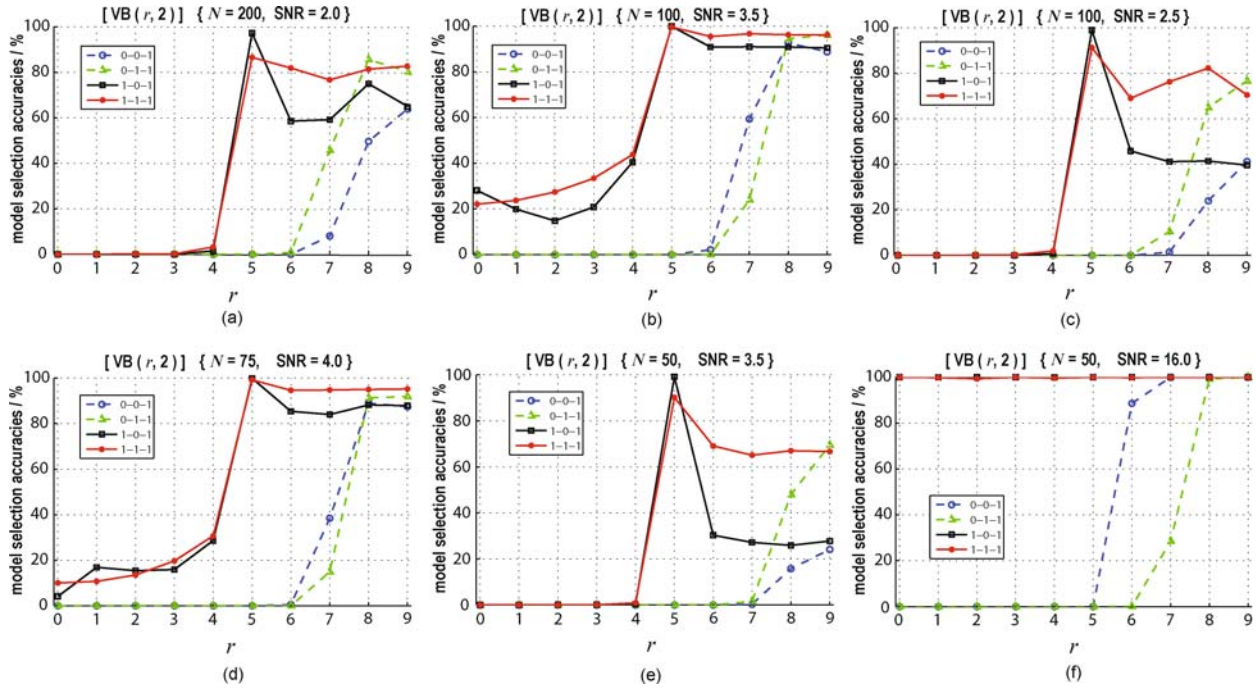


Fig. 5 Model selection accuracies of VB learning on FA-r against r. (Each curve represents a configuration of the priors on $(\boldsymbol{V}, \boldsymbol{\nu}, \varphi)$ in Table 3. For example, "0-0-1" denotes the configuration without priors on $\boldsymbol{V} = [\boldsymbol{A}, \boldsymbol{U}]$ or $\boldsymbol{\nu}$, with priors on φ, and so on and so forth. VB is run at its best implementation level, i.e., Level 2 (with the hyperparameters fixed at the constants as used in Refs. [8,26]), which has been shown to be the best in Fig. 3. Note that FA-r is FA-a at $r = 0$, or FA-b at $r = 9$ (because the used maximum candidate scale is 9).)

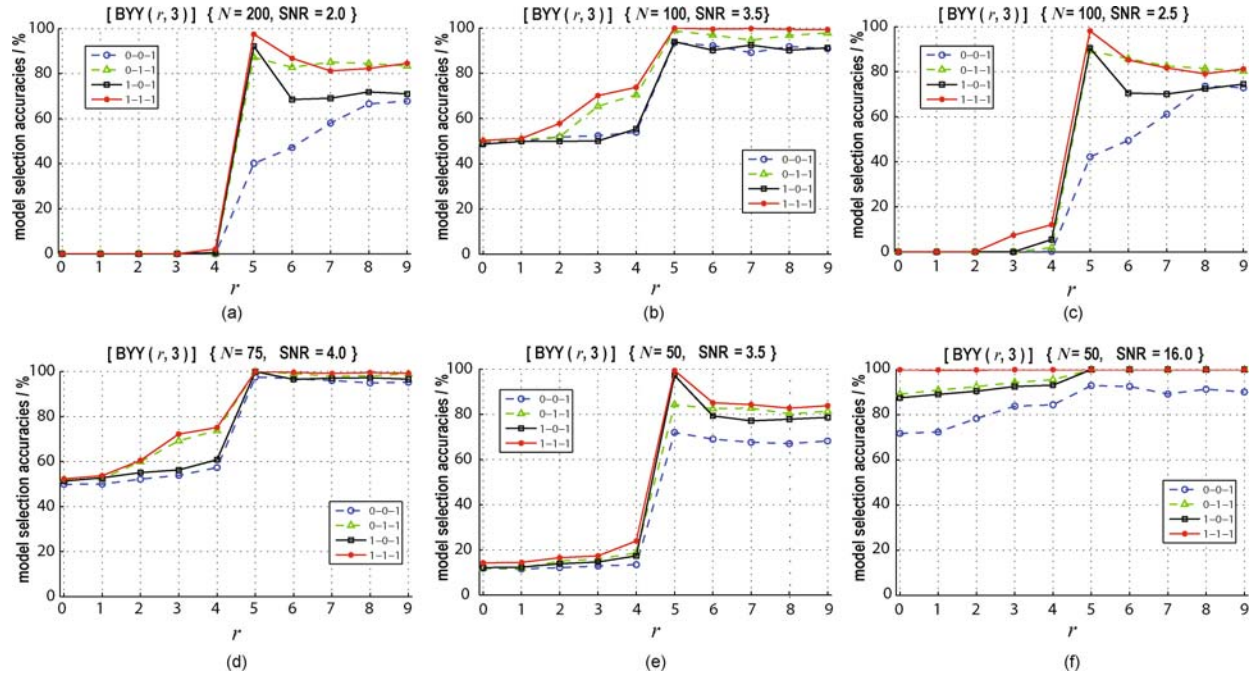


Fig. 6 Experimental results on FA-r with different parts of priors under BYY which runs at Level 3 (i.e., with the hyperparameters updated during learning). (Refer to the caption of Fig. 5 for the notation details.)

FA-r with $r \geqslant m^*$ than with $r < m^*$.

It can be observed from Fig. 5 that VB's model selection performance highly depends on the presence of the prior $q(\boldsymbol{V}) = q(\boldsymbol{A})q(\boldsymbol{U})$ when $r \leqslant m^*$, and the presence of the prior $q(\boldsymbol{\nu})$ when $r > m^*$, whereas this dependence is significantly weakened as FA-r becomes close to FA-b. Moreover, the prior $q(\varphi)$ is able to individually contribute a large proportion to model selection for a large enough r, especially when N and SNR are relatively large. That is, FA-b is better than FA-a under VB even if only one and same prior $q(\varphi)$ is considered.

As shown in Fig. 6, BYY's model selection performance does not reply on priors as much as VB, because BYY is capable of model selection even without any priors (see the results of BYY(a,1) and BYY(b,1) in Fig. 4 or refer to Fig. 5 in Sect. 2.2 of Ref. [22] for a detailed explanation). Moreover, BYY works slightly better with $q(\boldsymbol{\nu})$ in place of $q(\boldsymbol{V})$.

The performance gain from the incorporation of the priors over the parameters becomes small as we proceed from FA-a to FA-b. Appropriate priors are very helpful for a not good model parameterization, but they may become not so critical if a good model parameterization is chosen. Moreover, it should be noted that the $q(\boldsymbol{U})$ given in Table 3 has no impact on parameter learning but only help model selection at Stage II.

6.3.2 r should be no less than needed

We further make empirical analysis on how model selection performances vary on FA-r as r changes, by VB and BYY with all priors used.

The experiments are still conducted on $\mathcal{S}(:,:,15,5)$. The model selection accuracies of VB on FA-r for different r are reported in terms of contour maps in Fig. 7.

The best performance is achieved on the parameterization FA-m^* ($m^* = 5$), which provides a correct calibration though this m^* is practically unknown. On one hand, the performances on those of FA-r drop sharply as r reduces from m^* towards FA-a, which implies that the parts of FA-a combined in FA-r make negative contributions to model selection. On the other hand, the performance on FA-r deteriorates slightly and slowly as r increases from m^* towards FA-b, though extra parameters in the covariance of the hidden variables incurs certain overfitting. The above characteristic is better demonstrated in Fig. 8.

The model selection accuracies by BYY on FA-r show a similar trend, and thus omitted here. Moreover, BYY outperforms VB for a large enough r or for a FA-r close to FA-b. Specifically, we compare their best cases (both at $r = 5$) among all FA-r in Fig. 9. BYY is better for

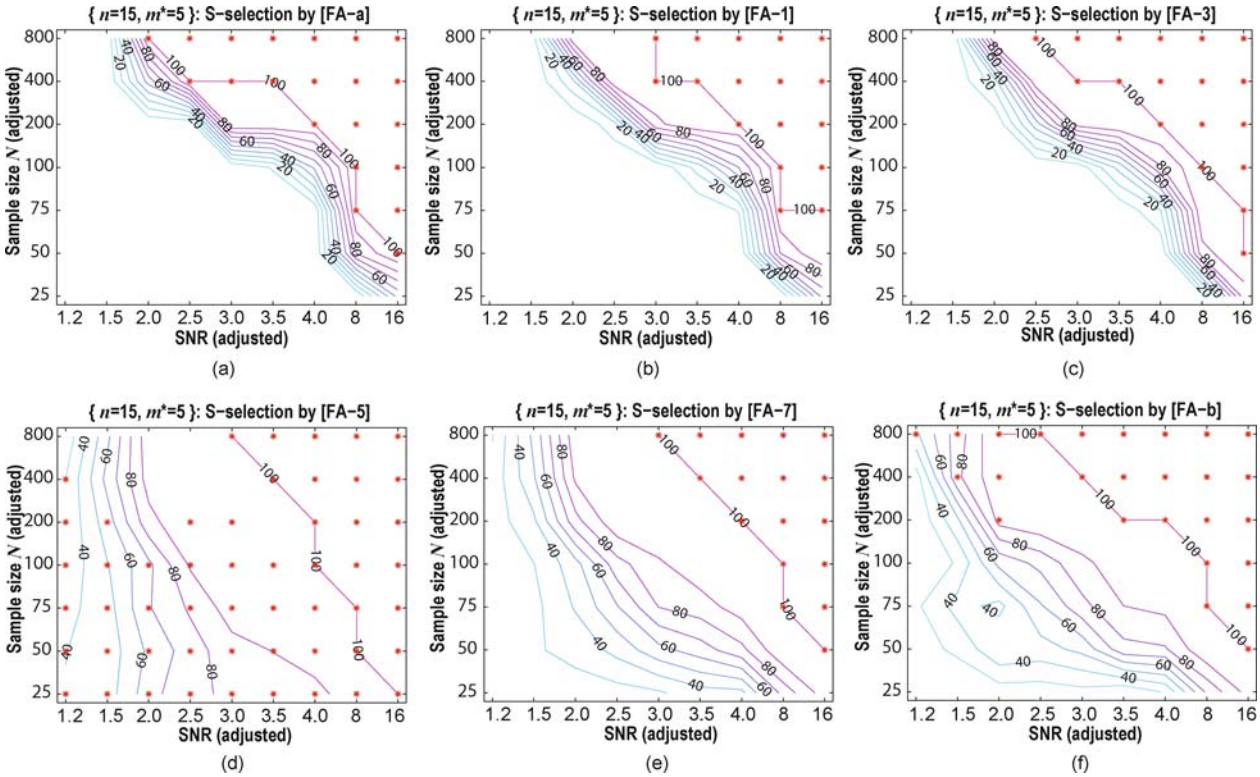


Fig. 7 Model selection accuracies of FA-r under VB for $r = 0, 1, 3, 5, 7, 9$, on the same synthetic data as Figs. 2 to 4. (A red asterisk ($*$) indicates a highest accuracy among all the subfigures in this figure. The results are obtained from Level 2 implementation in Table 7, because VB(b,2) is the best in Fig. 3.)

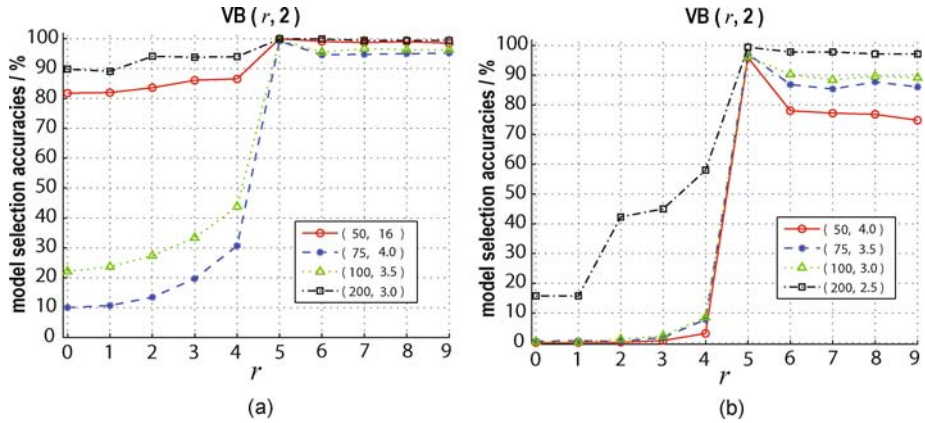


Fig. 8 Model selection accuracies of VB on FA-r for $r \in \{1, 2, \ldots, 9\}$ on different experimental settings (N, γ_o), i.e., $\mathcal{S}(N, \gamma_o, 15, 5)$, where N and γ_o denote sample size and SNR, respectively

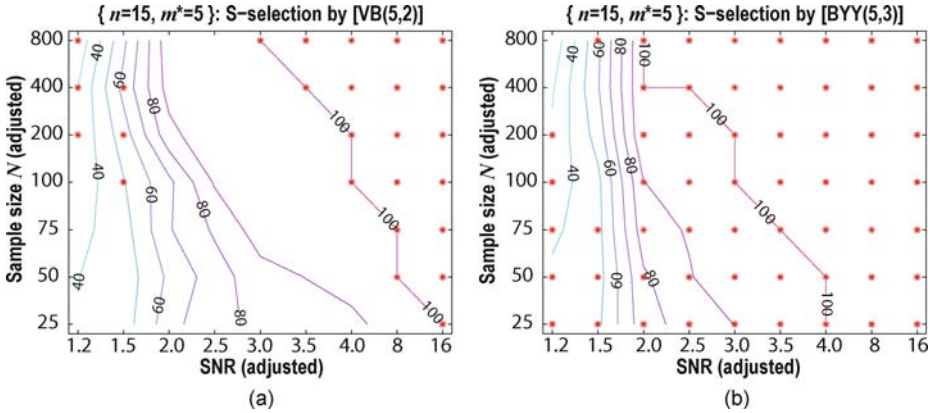


Fig. 9 A comparison of model selection performances of VB and BYY learning on FA-r at $r = m^* = 5$. (A red asterisk $(*)$ indicates a highest accuracy between VB(5,2) (a) and BYY(5,3) (b).)

most settings while VB is relatively better for a large N but small SNR. This observation is consistent with Figs. 3 and 4.

In fact, the above obtained trend has also been observed in Figs. 5 and 6 under different combinations of parts of priors on the parameters. Therefore, this trend is implied to be an intrinsic characteristic of FA-r in model selection.

Figure 10 visualizes how the values of VB's variational lower bound or BYY's harmony functional vary as r changes. The curves illustrate a "combined effect" of FA-a and FA-b on FA-r, i.e., they are approximately bounded between FA-a and FA-b. Thus, FA-r can robustly balance the underestimation and overestimation at $r = m^*$.

In summary, the parameterization family FA-r serves as a transition map (from FA-a to FA-b), based on which we are able to systematically study the various aspects of model selection performances. In this map, the FA-m^* is the best. Since m^* is unknown to be sought, we should use FA-r with a r large enough such that it is not less than m^*. Since the performance on FA-r deteriorates slightly and slowly as r increases from m^* towards FA-b, FA-b is a good alternative of FA-r with a right r, especially for BYY. The superiority of FA-b over FA-a is significant and reliable. With FA-a replaced by FA-b, the gain obtained by BYY is obviously higher than that by VB, while the gain by VB is better than no gain by AIC, BIC and DNLL, especially for a finite sample size.

6.4 FA-a versus FA-b: Automatic model selection performance of BYY and VB

The above investigations are all based on two-stage procedures. However, not only the two-stage procedure is computationally expensive, but also the performance of parameter estimation deteriorates for those candidate models with a large hidden dimensionality m (see Sect. 2.1 of Ref. [22]). Automatic model selection, i.e., discarding extra hidden dimensions during parameter learning, is one road to tackle the problems of two-stage implementation. Both BYY [22,27] and VB [8] have been found to be capable of automatic model selection. In this section, we investigate the automatic model selection performances of BYY and VB on both FA-a and FA-b.

During parameter learning by merely Stage I, we discard the ith hidden dimension if either of the following two equations hold:

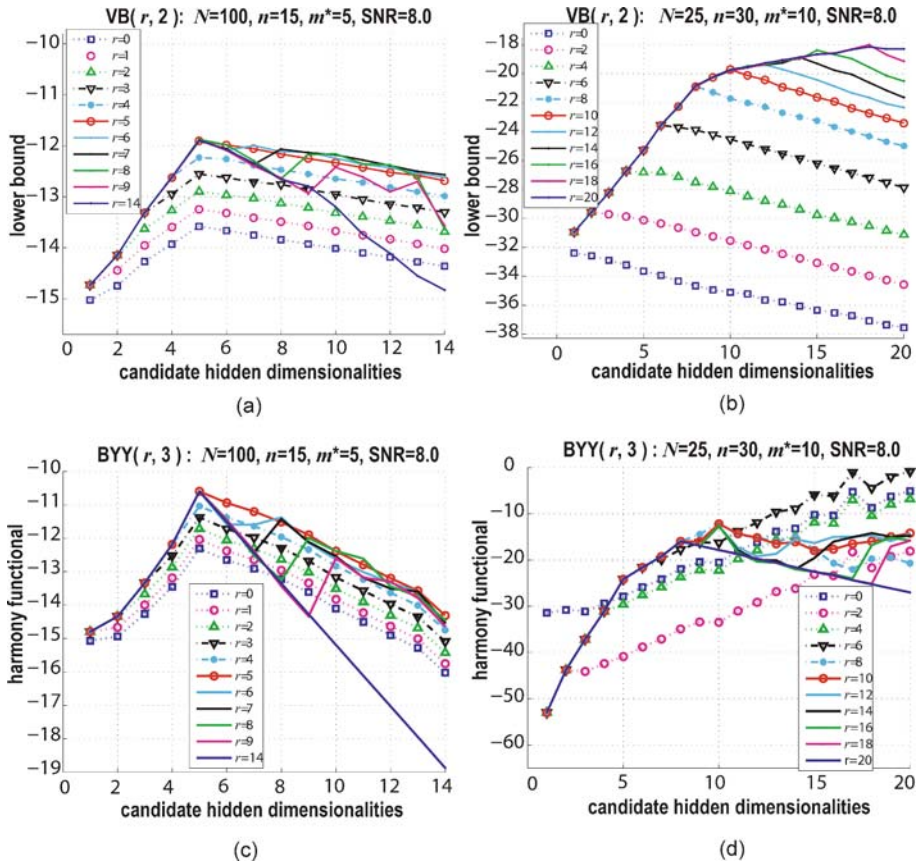


Fig. 10 Values of VB lower bound $\mathcal{F}$ or BYY harmony measure $\mathcal{H}$ on FA-r for two different settings $\mathcal{S}(N, 8.0, n, m^*)$. (The values of VB's variational lower bound $\mathcal{F}$ and BYY's harmony functional $\mathcal{H}$ are the largest one of 10 independently repeated trials on a synthetic data set randomly generated according to $\mathcal{S}(N, 8.0, n, m^*)$. The axis represents the candidate hidden dimensionality $m \in \mathcal{M}$, where $\mathcal{M} = \{1, 2, \ldots, 14\}$ when $n = 15$; otherwise $\mathcal{M} = \{1, 2, \ldots, 20\}$. Specifically, FA-$r$ becomes FA-a when $r = 0$, FA-b when $r = 14$ for (a)(c) and $r = 20$ for (b)(d) respectively.)

$$1/\alpha_i < \eta_0, \text{ for FA-a;} \quad 1/\nu_i < \eta_0, \text{ for FA-b,} \tag{26}$$

$$\mathcal{J}^{(\tau_2)}(m-1) \geqslant \mathcal{J}^{(\tau_1)}(m), \quad \mathcal{J} \text{ is given by}$$
$$\text{Eq. (9) for VB, Eq. (19) for BYY,} \tag{27}$$

where τ_1, τ_2 are iteration indicators meaning the τ_1th, τ_2th iteration respectively, subject to

1) $|\mathcal{J}^{(k)} - \mathcal{J}^{(k-1)}| < \eta_1 \cdot |\mathcal{J}^{(k)}|$, where $\mathcal{J}^{(k)}$ is the value of the learning objective function evaluated at the kth step, and $k = \tau_1, \tau_2$, $\tau_1 < \tau_2$;

2) The hidden dimensionality is equal to m at the τ_1th step, and is reduced to be $m-1$ at the (τ_1+1)th step, with the ith dimension is temporarily discarded, where $i = \arg\min_j\{1/\alpha_j\}$ for FA-a, $i = \arg\min_j\{1/\nu_j\}$ for FA-b.

Since an arbitrary value for η_0 in Eq. (26) for all algorithms may not be good and fair, an alternative way in Eq. (27) is to check whether a reduction brings down the value of the objective function.

In the experiments on the settings $S(:, :, 15, 5)$, we set $\eta_0 = 0.01$, $\eta_1 = 0.001$, and initialize $m = 9$. We report the automatic model selection accuracies of VB(a,3), VB(b,2), BYY(a,3), and BYY(b,3) in Fig. 11, because they are the best implementation levels respectively according to Figs. 3 and 4. The results again show that FA-b is better than FA-a, and BYY(b,3) is superior for a small sample size, while VB(b,2) has the advantage when the sample size is large and SNR is small. As a whole, automatic model selection performances are not so good as those by two-stage implementation, because Eq. (26) relies on parameter learning without using the terms only related to m, and Eq. (27) is a simple depth-first search without evaluating the objective function for all candidate dimensionalities. Moreover, VB is not good for automatic model selection under Eqs. (26) and (27) for its automatic performance deteriorates more rapidly than that of BYY does.

6.5 Classification performance on real world data sets

We consider two real world data sets: Pendigits (PEN) and Segment (SEG), taken from the UCI machine learning repository (http://archive.ics.uci.edu/ml/datasets.html) [32]. For each out of 100 independent runs, a training set of a chosen sample size $N \in \{16, 20, 30, 80\}$ is made up of instances randomly picked from the original data set, while the rest instances are put in a testing set. The details are listed in Table 9. The candidate scale set for the two-stage procedure is $\mathcal{M} = \{1, 2, \ldots, 15\}$. We

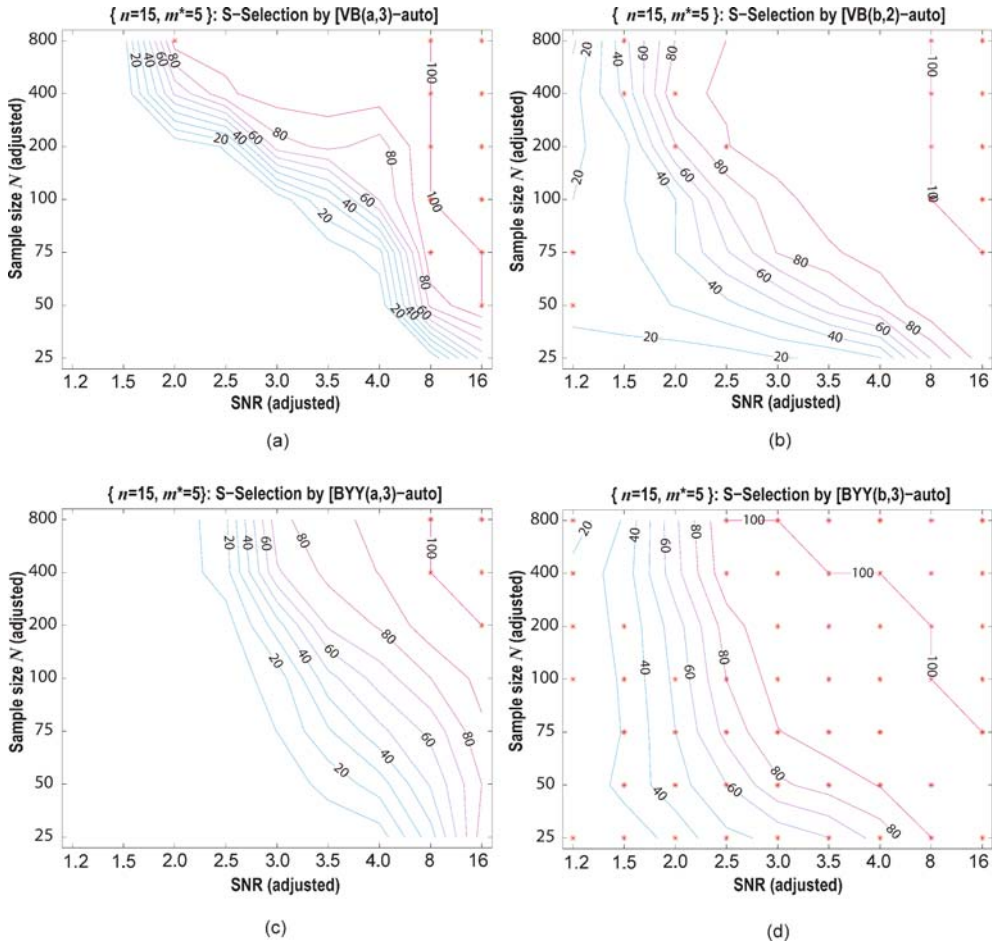


Fig. 11 Automatic model selection accuracies of VB and BYY on FA-a and FA-b ($n = 15, m^* = 5$)

estimate a model for every class, and get classification accuracies on testing sets by a Bayesian classifier.

Table 9 Real world data sets: Pendigits (PEN) and Segment (SEG) (PEN consists of 16 attributes, 10 classes, 10992 instances; SEG consists of 16 attributes, 7 classes, 2310 instances. Here, the SEG is preprocessed by a normalization and discarding three attributes, i.e., $(3, 4, 5)$, in the original 19 attributes.)

data set	training size	testing size	data set	training size	testing size
PEN-16	16×10	10832	SEG-16	16×7	2198
PEN-20	20×10	10792	SEG-20	20×7	2170
PEN-30	30×10	10692	SEG-30	30×7	2100
PEN-80	80×10	10192	SEG-80	80×7	1750

The results are reported in Table 10. Most criteria achieve comparable performances when the sample size is large, and deteriorate as the sample size reduces, which is consistent with Figs. 3 and 4. BYY is better than VB for most cases, and is comparable to VB for the rest cases but still preferred because of a smaller standard deviation. It can be observed that the recognition rates in Table 10 are different from the model selection rates. A correct model selection helps to improve classification accuracy. However, an oversized model may not considerably deteriorate the classification accuracy, depending on the nature of tasks, e.g., if an extra dimension does no harm to generalization, the influence of model selection on classification may not be very obvious.

7 Concluding remarks

Focusing on FA, we have made a systematic empirical investigation on how parameterizations affect model selection performance, which was an issue that has been ignored or seldom studied. To this purpose, we present a new family of FA parameterizations, FA-r, that have equivalent likelihood functions with FA-a and FA-b as two ends, where FA-a and FA-b were previously known to be different in model selection under BYY.

Several empirical finds have been obtained via extensive experiments. First, both BYY and VB perform obviously better on FA-b than on FA-a. Specially, both BYY and VB achieve their best performances on the parameterization FA-m^* with m^* being the correct number of hidden factors. The performance on those FA-r close to FA-b is considerably superior to those FA-r close to FA-a. Since m^* is unknown to be sought, we should use FA-r with an r large enough such that it is not smaller than m^*. Due to the performance on FA-r deteriorates

Table 10 Experiment results on real world data sets PEN and SEG (The classification accuracies (%) are reported in the form of "*average*±*standard deviation*" by 100 independent runs for each setting. The improvement from a higher level implementation (except VB on FA-b) reduces as the sample size grows. Moreover, the improvements from BYY(b,1) to BYY(b,2), and then to BYY(b,3) are small because BYY(b,1) performs already very good, especially for a small sample size.)

criteria	PEN-16	PEN-20	PEN-30	PEN-80
VB(a,2)	85.44±1.07	89.96±1.14	92.90±0.98	95.99±0.50
VB(a,3)	86.82±1.53	91.02±1.19	93.11±1.01	96.02±0.55
VB(b,2)	87.02±2.45	91.68±1.81	93.86±1.31	96.09±0.31
VB(b,3)	84.87±1.78	90.42±0.09	93.01±1.34	96.01±0.41
BYY(a,1)	55.25±6.40	64.13±6.22	71.77±6.42	84.70±4.17
BYY(a,2)	82.77±2.04	83.52±2.23	84.54±2.02	85.63±1.91
BYY(a,3)	87.26±1.30	88.13±1.24	90.65±1.86	89.14±1.57
BYY(b,1)	87.31±1.08	93.31±1.03	93.71±0.37	95.95±0.41
BYY(b,2)	87.01±1.06	93.27±0.72	94.01±0.21	96.16±0.19
BYY(b,3)	**88.57**±1.04	**93.55**±0.51	**94.15**±0.33	**96.17**±0.22
criteria	SEG-16	SEG-20	SEG-30	SEG-80
VB(a,2)	71.57±4.12	78.06±3.33	88.02±2.35	97.57±0.32
VB(a,3)	72.73±3.23	77.78±2.78	82.63±2.02	97.22±1.25
VB(b,2)	75.98±5.03	79.48±2.99	**89.13**±1.65	97.61±1.06
VB(b,3)	69.73±6.76	75.49±6.10	87.15±1.24	97.27±0.49
BYY(a,1)	79.40±2.71	80.43±3.23	77.25±4.21	78.71±4.71
BYY(a,2)	68.69±5.37	79.98±5.14	78.95±4.74	81.15±4.37
BYY(a,3)	82.02±3.02	84.09±2.87	85.91±2.73	86.78±2.47
BYY(b,1)	82.14±2.08	85.17±1.60	87.91±1.93	97.56±0.37
BYY(b,2)	84.87±1.50	85.65±1.42	88.32±1.31	97.62±0.51
BYY(b,3)	**85.01**±0.98	**85.91**±0.91	89.01±1.19	**97.65**±0.31

slightly and slowly as r increases from m^* towards FA-b, FA-b is a good alternative of FA-r with a right r, especially for BYY. The superiority of FA-b over FA-a is significant and reliable. Second, both BYY and VB outperform AIC, BIC, and DNLL, while BYY further outperforms VB, especially on FA-b. Moreover, BYY obtains a higher gain than VB does on FA-b in place of FA-a, while the gain by VB is better than no gain by AIC, BIC, and DNLL, especially when the sample size is small. Third, we have also investigated how each part of the priors contributes to the model selection performance, and found that appropriate priors are beneficial in model selection, but BYY does not highly depend on the presences of the priors whereas VB does. Moreover, optimizing hyperparameters further improve the performance of BYY whereas it deteriorates the performance of VB, which indicates that a good learning approach weakens its dependence on getting appropriate priors to obtain improved performance.

The above empirical findings on the superiority of FA-b over FA-a concur with two recent analytical justifications on the superiority of FA-b over FA-a made in Ref. [28] (especially see the paragraph around its Eq. (28)). That is, the FA-b with $G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Lambda})$ is better than the FA-a with $G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{I})$ in term of providing one additional room for model selection either directly via $\boldsymbol{\Lambda}$ by BYY or via adding priors by BYY, VB and also other Bayesian approaches. In comparison with getting priors, $G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Lambda})$ is more reliable and easy to be estimated from data. Moreover, more reliable information about m is contained in $G(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{\Lambda})$ than in priors. Thus, BYY can considerably outperforms VB on FA-b.

Appendix A VB learning algorithm on FA-r

According to e.g., Eq. (13) in Ref. [8], or Theorem 2.1 in Ref. [21], the variational posteriors are iteratively derived by optimizing the variational lower bound $\mathcal{F}$ in Eq. (7) with the other variational posteriors fixed. Mathematically, we have

$$p_Y \propto \exp\left\{\int \ln[q(X_N, Y|\boldsymbol{\Theta})q(\boldsymbol{\Theta})]\,\mathrm{d}\boldsymbol{\Theta}\right\}, \tag{A.1}$$

$$p_{\boldsymbol{\theta}_j} \propto \exp\left\{\int \ln[q(X_N, Y|\boldsymbol{\Theta})q(\boldsymbol{\Theta})]\,\mathrm{d}Y\mathrm{d}\boldsymbol{\theta}_{i\neq j}\right\}, \tag{A.2}$$

where $p_{\boldsymbol{\Theta}} = \prod_i p_{\boldsymbol{\theta}_i}$ and $\boldsymbol{\Theta} = \{\boldsymbol{\theta}_i, \forall i\}$, $\mathrm{d}\boldsymbol{\theta}_{i\neq j} = \mathrm{d}\boldsymbol{\theta}_1\mathrm{d}\boldsymbol{\theta}_2\cdots\mathrm{d}\boldsymbol{\theta}_{i-1}\mathrm{d}\boldsymbol{\theta}_{i+1}\cdots\mathrm{d}\boldsymbol{\theta}_u$.

It follows from the variational lower bound $\mathcal{F}$ on FA-r by Eq. (11) that we can derive the following variational posteriors:

$$p_Y = \prod_{t=1}^{N} G(\boldsymbol{y}_t|\boldsymbol{\mu}_{y|x}^{(t)}, \boldsymbol{\Sigma}_{y|x}) \propto \exp\left\{\sum_{t=1}^{N}\left\{-\frac{1}{2}\boldsymbol{y}_t^{\mathrm{T}}\left(\mathrm{E}[\boldsymbol{V}^{\mathrm{T}}\boldsymbol{\varphi}\boldsymbol{V}] + \mathrm{diag}[\boldsymbol{\mu}_\nu, \boldsymbol{I}_{m-r}]\right)\boldsymbol{y}_t + \boldsymbol{y}_t^{\mathrm{T}}\boldsymbol{\mu}_V^{\mathrm{T}}\boldsymbol{\mu}_{\boldsymbol{\varphi}}\boldsymbol{x}_t\right\}\right\},$$

$$p_A = \prod_{j=1}^{n} G(\boldsymbol{a}_j|\boldsymbol{\mu}_{A,j}, \boldsymbol{\Sigma}_{A,j}) \propto \exp\left\{\sum_{j=1}^{n}\left\{-\frac{1}{2}\boldsymbol{a}_j^{\mathrm{T}}\left(\boldsymbol{\mu}_{\varphi_j}\sum_{t=1}^{N}\mathrm{E}\left[\boldsymbol{y}_t^{(m-r)}\boldsymbol{y}_t^{(m-r)\mathrm{T}}\right] + \mathrm{diag}[\boldsymbol{\mu}_{\boldsymbol{\alpha}}]\right)\boldsymbol{a}_j^{\mathrm{T}}\right.\right.$$
$$\left.\left. + \boldsymbol{a}_j^{\mathrm{T}}\left(\boldsymbol{\mu}_{\varphi_j}\sum_{t=1}^{N}\mathrm{E}\left[(\boldsymbol{x}_{jt} - \sum_{i=1}^{r} y_{it}u_{ji})\boldsymbol{y}_t^{(m-r)}\right]\right)\right\}\right\},$$

$$p_{\boldsymbol{\alpha}} = \prod_{k=1}^{m-r}\Gamma\left(\alpha_k|\hat{a}_k^{\alpha}, \hat{b}_k^{\alpha}\right) \propto \exp\left\{\sum_{k=1}^{m-r}\left(\alpha_k^{\alpha} + \frac{n}{2} - 1\right)\ln\alpha_k - \sum_{k=1}^{m-r}\left(b_k^{\alpha} + \frac{\mathrm{E}[\boldsymbol{a}_k^{\mathrm{T}}\boldsymbol{a}_k]}{2}\right)\alpha_k\right\},$$

$$p_U \approx \delta(\boldsymbol{U}-\boldsymbol{U}_S^*),\ \ \boldsymbol{U}_S^*=\boldsymbol{U}_E^*(\boldsymbol{U}_E^{*\mathrm{T}}\boldsymbol{U}_E^*)^{-1/2},$$

$$\boldsymbol{U}_E^*=\left(\sum_{t=1}^N \boldsymbol{x}_t \mathrm{E}\left[\boldsymbol{y}_t^{(r)\mathrm{T}}\right]-\boldsymbol{\mu}_A\sum_{t=1}^N \mathrm{E}\left[\boldsymbol{y}_t^{(m-r)}\boldsymbol{y}_t^{(r)\mathrm{T}}\right]\right)\left(\sum_{t=1}^N \mathrm{E}\left[\boldsymbol{y}_t^{(r)}\boldsymbol{y}_t^{(r)\mathrm{T}}\right]\right)^{-1},$$

$$p_\nu=\prod_{i=1}^r \Gamma\left(\nu_i|\hat{a}_i^\nu,\hat{b}_i^\nu\right)\propto \exp\left\{\sum_{i=1}^r\left(a_i^\nu+\frac{N}{2}-1\right)\ln\nu_i-\sum_{i=1}^r\left(b_i^\nu+\frac{1}{2}\sum_{t=1}^N \mathrm{E}\left[\boldsymbol{y}_t^{(r)}\boldsymbol{y}_t^{(r)\mathrm{T}}\right]_{(i,i)}\right)\nu_i\right\},$$

$$p(\boldsymbol{\varphi})=\Gamma\left(\varphi|\hat{a}_j^\varphi,\hat{b}_j^\varphi\right)\propto \exp\left\{\sum_{j=1}^n\left(a_j^\varphi+\frac{N}{2}-1\right)\ln\varphi-\sum_{j=1}^n\left(b_j^\varphi+\frac{1}{2}\sum_{t=1}^N \mathrm{E}\left[(x_{jt}-\boldsymbol{v}_j^{\mathrm{T}}\boldsymbol{y}_t)^2\right]\right)\varphi\right\},$$

where all the notations are referred to Table A1, which summarizes the detailed iterations of the VB learning algorithm.

Table A1 Details of the VB learning algorithm on FA-r ($\mathrm{E}_p[f(\theta)]$ denotes the expectation of $f(\theta)$ with respect to (variational posterior) $p(\theta)$, and the subscript p is omitted for notation simplicity.)

objective: maximize the variational lower bound

$\mathcal{F}(p_{\boldsymbol{\Theta}},p_Y,\Xi_r)=\int p_A p_U p_\nu p_\varphi p_Y \ln\left[\prod_{t=1}^N\{G(\boldsymbol{x}_t|\boldsymbol{V}\boldsymbol{y}_t,\varphi^{-1}\boldsymbol{I}_n)G(\boldsymbol{y}_t|\boldsymbol{0},\boldsymbol{\Sigma}_y^r)\}\prod_{i=1}^r\Gamma(\nu_i|a_i^\nu,b_i^\nu)\right.$
$\left.\cdot\prod_{k=1}^{m-r}G(\boldsymbol{a}_k|\boldsymbol{0},\alpha_k^{-1}\boldsymbol{I}_n)\cdot q(\boldsymbol{U})\Gamma(\varphi|a^\varphi,b^\varphi)/(p_A p_U p_\nu p_\varphi p_Y)\right]\mathrm{d}Y\mathrm{d}\boldsymbol{A}\mathrm{d}\boldsymbol{U}\mathrm{d}\boldsymbol{\nu}\mathrm{d}\varphi$

initialization: initialize p_θ, $\boldsymbol{\theta}\in\{\boldsymbol{U},\boldsymbol{A},\boldsymbol{\nu},\boldsymbol{\alpha},\varphi\}$, $\boldsymbol{\varphi}=\varphi\boldsymbol{I}_n$, $\boldsymbol{V}=[\boldsymbol{U},\boldsymbol{A}]$,

update variational posteriors: Calculate p_{θ_i} based on $p_{\theta_j},\forall j\neq i$, where $\theta_\ell\in\{Y,\boldsymbol{A},\boldsymbol{U},\boldsymbol{\alpha},\boldsymbol{\nu},\varphi\}$.

- Calculate p_Y by $\boldsymbol{\Sigma}_{y|x}=\left(\mathrm{E}[\boldsymbol{V}^{\mathrm{T}}\varphi\boldsymbol{V}]+\mathrm{diag}[\boldsymbol{\mu}_\nu,\boldsymbol{I}_{m-r}]\right)^{-1}$, $\boldsymbol{\mu}_{y|x}^{(t)}=\boldsymbol{\Sigma}_{y|x}\boldsymbol{\mu}_V^{\mathrm{T}}\boldsymbol{\mu}_\varphi\boldsymbol{x}_t$, where

  $\mathrm{E}[\boldsymbol{V}^{\mathrm{T}}\varphi\boldsymbol{V}]=\mathrm{E}\{[\boldsymbol{U},\boldsymbol{A}]^{\mathrm{T}}\varphi[\boldsymbol{U},\boldsymbol{A}]\}=\begin{pmatrix}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{\mu}_\varphi\boldsymbol{U} & \boldsymbol{U}^{\mathrm{T}}\boldsymbol{\mu}_\varphi\boldsymbol{\mu}_A\\ \boldsymbol{\mu}_A^{\mathrm{T}}\boldsymbol{\mu}_\varphi\boldsymbol{U} & \mathrm{E}[\boldsymbol{A}^{\mathrm{T}}\varphi\boldsymbol{A}]\end{pmatrix}$, $\boldsymbol{\mu}_V=[\boldsymbol{U},\boldsymbol{\mu}_A]$, $\boldsymbol{\mu}_\varphi=\mathrm{diag}[\mu_{\varphi_1},\mu_{\varphi_2},\ldots,\mu_{\varphi_n}]$,

  $\mathrm{E}[\boldsymbol{A}^{\mathrm{T}}\varphi\boldsymbol{A}]=\sum_{j=1}^n\boldsymbol{\mu}_{\varphi_j}\mathrm{E}[\boldsymbol{a}_j\boldsymbol{a}_j^{\mathrm{T}}]=\sum_{j=1}^n\boldsymbol{\mu}_{\varphi_j}(\boldsymbol{\Sigma}_{A,j}+\boldsymbol{\mu}_{A,j}\boldsymbol{\mu}_{A,j}^{\mathrm{T}})$, $\quad\boldsymbol{a}_j=j$th row of $\boldsymbol{A}$.
- Calculate p_A by $\boldsymbol{\Sigma}_{A,j}=\left(\boldsymbol{\mu}_{\varphi_j}\sum_{t=1}^N\mathrm{E}[\boldsymbol{y}_t^{(m-r)}\boldsymbol{y}_t^{(m-r)\mathrm{T}}]+\mathrm{diag}[\boldsymbol{\mu}_{\boldsymbol{\alpha}}]\right)^{-1}$,

  $\boldsymbol{\mu}_{A,j}=\boldsymbol{\Sigma}_{A,j}\boldsymbol{\mu}_{\varphi_j}\sum_{t=1}^N\mathrm{E}\left[(x_{jt}-\sum_{i=1}^r y_{it}u_{ji})\boldsymbol{y}_t^{(m-r)}\right]$,

  $\sum_{t=1}^N\mathrm{E}[\boldsymbol{y}_t\boldsymbol{y}_t^{\mathrm{T}}]=\begin{pmatrix}\sum_t\mathrm{E}[\boldsymbol{y}_t^{(r)}\boldsymbol{y}_t^{(r)\mathrm{T}}] & \sum_t\mathrm{E}[\boldsymbol{y}_t^{(r)}\boldsymbol{y}_t^{(m-r)\mathrm{T}}]\\ \sum_t\mathrm{E}[\boldsymbol{y}_t^{(m-r)}\boldsymbol{y}_t^{(r)\mathrm{T}}] & \sum_t\mathrm{E}[\boldsymbol{y}_t^{(m-r)}\boldsymbol{y}_t^{(m-r)\mathrm{T}}]\end{pmatrix}=N\boldsymbol{\Sigma}_{y|x}+\sum_{t=1}^N\boldsymbol{\mu}_{y,t}\boldsymbol{\mu}_{y,t}^{\mathrm{T}}$,

  $\sum_{t=1}^N\mathrm{E}\left[(\boldsymbol{x}_{jt}-\sum_{i=1}^r y_{it}u_{ji})\boldsymbol{y}_t^{(m-r)}\right]=\mathrm{E}[Y^{(m-r)}]X_{j\bullet}^{\mathrm{T}}-\left(\sum_{t=1}^N\mathrm{E}\left[\boldsymbol{y}_t^{(m-r)}\boldsymbol{y}_t^{(r)\mathrm{T}}\right]\right)\boldsymbol{U}_{j\bullet}^{\mathrm{T}}$,

  $\boldsymbol{\mu}_{y|x}^{(t)}=(\mathrm{E}[\boldsymbol{y}_t^{(r)}];\mathrm{E}[\boldsymbol{y}_t^{(m-r)}])$, $Y=[y_{it}]_{m\times N}$, $\mathrm{E}[Y^{(r)}]=[\mathrm{E}(\boldsymbol{y}_t^{(r)})]_{r\times N}$, $Y^{(m-r)}=[\boldsymbol{y}_t^{(m-r)}]_{(m-r)\times N}$,

  $X=[x_{jt}]_{n\times N}$, $\ X_{j\bullet}=[x_{j1},x_{j2},\ldots,x_{jN}]_{1\times N}$, $\boldsymbol{U}=[u_{ji}]_{n\times r}$, $\ \boldsymbol{u}_j^{\mathrm{T}}=\boldsymbol{U}_{j\bullet}=[u_{j1},u_{j2},\ldots,u_{jr}]_{1\times r}$, $\boldsymbol{v}_j=[\boldsymbol{u}_j,\boldsymbol{a}_j]$.
- calculate $p_U\approx\delta(\boldsymbol{U}-\boldsymbol{U}_S^*)$, $\boldsymbol{U}_S^*=\boldsymbol{U}_E^*\left(\boldsymbol{U}_E^{*\mathrm{T}}\boldsymbol{U}_E^*\right)^{-\frac{1}{2}}$, $\boldsymbol{U}_E^*=\left(\sum_t\boldsymbol{x}_t(\boldsymbol{\mu}_{y|x}^{(t)})^{\mathrm{T}}\right)\left(\sum_t\mathrm{E}[\boldsymbol{y}_t\boldsymbol{y}_t^{\mathrm{T}}]\right)^{-1}$.
- calculate p_α by $\hat{a}_k^\alpha=a_k^\alpha+\frac{n}{2}$, $\hat{b}_k^\alpha=b_k^\alpha+\frac{\mathrm{E}[\boldsymbol{a}_k^{\mathrm{T}}\boldsymbol{a}_k]}{2}$, $\boldsymbol{\mu}_{\boldsymbol{\alpha}}=[\mu_{\alpha_1},\mu_{\alpha_2},\ldots,\mu_{\alpha_{m-r}}]$, $\mu_{\alpha_k}=\hat{a}_k^\alpha/\hat{b}_k^\alpha$.

  $\mathrm{E}[\boldsymbol{a}_k^{\mathrm{T}}\boldsymbol{a}_k]=\left(\sum_{j=1}^n\boldsymbol{\Sigma}_{A,j}+\boldsymbol{\mu}_{A,j}\boldsymbol{\mu}_{A,j}^{\mathrm{T}}\right)_{(k,k)}$, $M_{(k,k)}$ denotes the kth diagonal element of M.
- calculate p_ν by $\hat{a}_i^\nu=a_i^\nu+\frac{N}{2}$, $\hat{b}_i^\nu=b_i^\nu+\frac{1}{2}\sum_{t=1}^N\mathrm{E}\left[\boldsymbol{y}_t^{(r)}\boldsymbol{y}_t^{(r)\mathrm{T}}\right]_{(i,i)}$, $\boldsymbol{\mu}_{\boldsymbol{\nu}}=[\mu_{\nu_1},\mu_{\nu_2},\ldots,\mu_{\nu_r}]$, $\mu_{\nu_i}=\hat{a}_i^\nu/\hat{b}_i^\nu$.
- Calculate p_φ by $\hat{a}^\varphi=a^\varphi+\frac{Nn}{2}$, $\hat{b}^\varphi=b^\varphi+\mathrm{Tr}[XX^{\mathrm{T}}]+\mathrm{Tr}\left\{\mathrm{E}[\boldsymbol{v}_j\boldsymbol{v}_j^{\mathrm{T}}]\sum_t\mathrm{E}[\boldsymbol{y}_t\boldsymbol{y}_t^{\mathrm{T}}]\right\}$, $\mu_{\varphi_j}=\hat{a}^\varphi/\hat{b}^\varphi$.

update hyperparameters Ξ by gradient method,

$a_k^{\alpha,\mathrm{new}}=\exp\{\xi_{k1}^{\mathrm{new}}\}$, $\xi_{k1}^{\mathrm{new}}=\xi_{k1}^{\mathrm{old}}+\eta\frac{\partial\mathcal{F}}{\partial\xi_{k1}}$, $\frac{\partial\mathcal{F}}{\partial\xi_{k1}}=a_k^\alpha(\ln b_k^\alpha-\psi(a_k^\alpha)+\mathrm{E}[\ln\alpha_k])$,

$b_k^{\alpha,\mathrm{new}}=\exp\{\xi_{k2}^{\mathrm{new}}\}$, $\xi_{k2}^{\mathrm{new}}=\xi_{k2}^{\mathrm{old}}+\eta\frac{\partial\mathcal{F}}{\partial\xi_{k2}}$, $\frac{\partial\mathcal{F}}{\partial\xi_{k2}}=b_k^\alpha(\frac{a_k^\alpha}{b_k^\alpha}-\boldsymbol{\mu}_{\alpha_k})$,

$\mathrm{E}[\ln\alpha_k]=\psi(\alpha_k)-\ln b_k^\alpha$, $\psi(\alpha_k)$: digamma function. Similar computation for $\{a_i^\nu,b_i^\nu,a^\varphi,b^\varphi\}$.

the details of $\mathcal{F}=\sum_{k=1}^{m-r}g(\alpha_k,a_k^\alpha,b_k^\alpha,p_\alpha)+\sum_{i=1}^r g(\nu_i,a_i^\nu,b_i^\nu,p_\nu)+g(\varphi,a^\varphi,b^\varphi,p_\varphi)$
$+\frac{Nn}{2}\mathrm{E}[\ln\varphi]-\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{\mu}_\varphi\sum_t\boldsymbol{x}_t\boldsymbol{x}_t^{\mathrm{T}}\right)+\mathrm{Tr}\left(\boldsymbol{\mu}_V^{\mathrm{T}}\boldsymbol{\mu}_\varphi\sum_t\boldsymbol{x}_t(\boldsymbol{\mu}_{y|x}^{(t)})^{\mathrm{T}}\right)-\frac{1}{2}\mathrm{Tr}\left(\mathrm{E}[\boldsymbol{V}^{\mathrm{T}}\varphi\boldsymbol{V}]\cdot\sum_t\mathrm{E}[\boldsymbol{y}_t\boldsymbol{y}_t^{\mathrm{T}}]\right)$
$-\frac{1}{2}\mathrm{Tr}\left\{\left(N\boldsymbol{\Sigma}_{y|x}+\sum_{t=1}^N\boldsymbol{\mu}_{y|x}^{(t)}\boldsymbol{\mu}_{y|x}^{(t)\mathrm{T}}\right)\cdot\mathrm{diag}[\boldsymbol{\mu}_{\boldsymbol{\nu}},\boldsymbol{I}_{m-r}]\right\}+\frac{Nm}{2}+\frac{N}{2}\ln\left|\mathrm{diag}[\boldsymbol{\mu}_{\boldsymbol{\nu}},\boldsymbol{I}_{m-r}]\cdot\boldsymbol{\Sigma}_{y|x}\right|$
$+\frac{n}{2}\sum_{k=1}^{m-r}\mathrm{E}[\ln\alpha_k]+\frac{1}{2}\sum_{j=1}^n\left\{\ln|\boldsymbol{\Sigma}_{A,j}|-\mathrm{Tr}\left[\left(\boldsymbol{\Sigma}_{A,j}+\boldsymbol{\mu}_{A,j}\boldsymbol{\mu}_{A,j}^{\mathrm{T}}\right)\cdot\mathrm{diag}[\boldsymbol{\mu}_{\boldsymbol{\alpha}}]\right]\right\}$
$+\frac{n(m-r)}{2}-\frac{Nn}{2}\ln(2\pi)-r\ln 2+\sum_{i=1}^r\left\{\ln\Gamma((n-i+1)/2)-\frac{n-i+1}{2}\ln\pi\right\}$,

where $g(z,a,b,p)=(a-\hat{a})\mathrm{E}_p[\ln z]-b\mathrm{E}_p[z]+\hat{a}+a\ln b-\hat{a}\ln\hat{b}-\ln\frac{\Gamma(a)}{\Gamma(\hat{a})}$, $p=\Gamma(z|\hat{a},\hat{b})$.

convergence: Repeat the above until the value of the variational lowerbound converges.

Appendix B BYY learning algorithm on FA-r

We adopt the gradient method to implement the Stage I in Table 5. The detailed harmony functional $H(p\|q)$ given in Eq.

(19) for FA-r consists of following five terms:

$$H(p\|q) \approx H_1 + d_r(\widetilde{\boldsymbol{W}}) + \ln q(\boldsymbol{\Theta}^a|\Xi) + H_b + \frac{1}{2}d_m(\Xi),$$

where the detailed formulas of the five terms are given in Eqs. (20) to (25). For each parameter $\boldsymbol{\theta} \in \{\boldsymbol{U}, \boldsymbol{A}, \boldsymbol{\nu}, \varphi\}$, or each hyperparameter $\xi \in \{a_k^\alpha, b_k^\alpha, a_i^\nu, b_i^\nu, a^\varphi, b^\varphi\}$, the gradients can be computed separately for each of the five terms. The computational details of the BYY learning algorithm are summarized in Table B1.

Table B1 Details of BYY learning algorithm on FA-r

objective: maximize the harmony functional: $H(p\|q) \approx H_1 + d_r(\widetilde{\boldsymbol{W}}) + \ln q(\boldsymbol{\Theta}^a|\Xi) + H_b + \frac{1}{2}d_{\mathbf{m}}(\Xi)$,
The first term is $H_1 = -\frac{N(n+m)}{2}\ln(2\pi) - \frac{Nm}{2} - \frac{1}{2}\mathrm{Tr}[\boldsymbol{S}_N\boldsymbol{\Sigma}_x^{-1}] + \frac{N}{2}\ln|\boldsymbol{\nu}_r| + \frac{N}{2}\ln|\varphi\boldsymbol{I}_n|$.
The last four terms are given by Eqs. (23), (24), (25), and (20).

initialization: randomly initialize $\boldsymbol{\theta} \in \{\boldsymbol{U}, \boldsymbol{A}, \boldsymbol{\nu}, \varphi\}$

gradient method: $\boldsymbol{\theta}^{\text{new}} = \boldsymbol{\theta}^{\text{old}} + \eta\partial\boldsymbol{\theta}$, $\partial\boldsymbol{\theta} = \partial^H\boldsymbol{\theta} = \frac{\partial H(p\|q)}{\partial\boldsymbol{\theta}}|_{\boldsymbol{\theta}=\boldsymbol{\theta}^{\text{old}}}$
$\partial\boldsymbol{\theta} = \partial^{H_1}\boldsymbol{\theta} + \partial^{d_r}\boldsymbol{\theta} + \partial^q\boldsymbol{\theta} + \partial^{H_b}\boldsymbol{\theta} + \partial^{d_m}\boldsymbol{\theta}$, from the five terms of $H(p\|q)$.

- $\partial\boldsymbol{a}_k = \partial^{H_1}\boldsymbol{a}_k + \partial^{d_r}\boldsymbol{a}_k + 0 + \partial^{H_b}\boldsymbol{a}_k + 0$, $\boldsymbol{a}_k$ is the kth column of $\boldsymbol{A}$.
  $\partial^{H_1}\boldsymbol{a}_k = \boldsymbol{M}\boldsymbol{V}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{a}_k}^{\mathrm{T}}$; $\partial^{H_b}\boldsymbol{a}_k = -(\hat{a}_k^\alpha - 1)(\hat{b}_k^\alpha)^{-1}\boldsymbol{a}_k$; $\partial^{d_r}\boldsymbol{a}_k = \boldsymbol{B}_w\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{a}_k}^{\mathrm{T}} - 2\boldsymbol{B}_w\boldsymbol{B}_x\mathcal{I}_{\boldsymbol{a}_k}^{\mathrm{T}} - \varphi\boldsymbol{V}\boldsymbol{\Delta}_w\boldsymbol{S}_N\boldsymbol{\Delta}_w^{\mathrm{T}}\mathcal{I}_{\boldsymbol{a}_k}^{\mathrm{T}}$.
- $\partial\boldsymbol{u}_i = \partial^{H_1}\boldsymbol{u}_i + \partial^{d_r}\boldsymbol{u}_i + 0 + 0 + 0$, $\boldsymbol{u}_i$ is the ith column of $\boldsymbol{U}$,
  $\partial^{H_1}\boldsymbol{u}_i = \boldsymbol{M}\boldsymbol{V}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{u}_i}$, $\partial^{d_r}\boldsymbol{u}_i = \boldsymbol{B}_w\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{u}_i}^{\mathrm{T}} - 2\boldsymbol{B}_w\boldsymbol{B}_x\mathcal{I}_{\boldsymbol{u}_i}^{\mathrm{T}} - \varphi\boldsymbol{V}\boldsymbol{\Delta}_w\boldsymbol{S}_N\boldsymbol{\Delta}_w^{\mathrm{T}}\mathcal{I}_{\boldsymbol{u}_i}^{\mathrm{T}}$.
- $\partial\nu_i = \partial^{H_1}\nu_i + \partial^{d_r}\nu_i + \partial^q\nu_i + 0 + 0$, $\boldsymbol{\nu} = \mathrm{diag}[\nu_1, \nu_2, \ldots, \nu_r]$,
  $\partial^{H_1}\nu_i = -\frac{1}{2}\mathrm{Tr}[\boldsymbol{B}_M\mathcal{I}_{\nu_i}] + \frac{N}{2}\nu_i^{-2}$, $\partial^q\nu_i = (a_i^\nu - 1)\nu_i^{-1} - b_i^\nu$, $\partial^{d_r}\nu_i = -\mathrm{Tr}[\boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}\boldsymbol{B}_w\boldsymbol{B}_x\mathcal{I}_{\nu_i}] - \frac{1}{2}\mathrm{Tr}[\boldsymbol{\Delta}_w\boldsymbol{S}_N\boldsymbol{\Delta}_w^{\mathrm{T}}\mathcal{I}_{\nu_i}]$.
- $\partial\varphi = \partial^{H_1}\varphi + \partial^{d_r}\varphi + \partial^q\varphi + 0 + 0$,
  $\partial^{H_1}\varphi = -\frac{1}{2}\mathrm{Tr}[\varphi^{-2}\boldsymbol{M}] + \frac{Nn}{2}\varphi^{-1}$, $\partial^q\varphi = (a^\varphi - 1)\varphi^{-1} - b^\varphi$, $\partial^{d_r}\varphi = \mathrm{Tr}[\varphi\boldsymbol{B}_w\boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}\boldsymbol{\Sigma}_x^{-1}\varphi] - \frac{1}{2}\mathrm{Tr}[\boldsymbol{V}\boldsymbol{\Delta}_w\boldsymbol{S}_N\boldsymbol{\Delta}_w\boldsymbol{V}^{\mathrm{T}}]$.

Hessian matrix $\Omega(\boldsymbol{\Theta}, \Xi)$ (approximated as block-diagonal):
$\partial^2\boldsymbol{a}_k = \boldsymbol{M}\mathcal{I}_{\boldsymbol{a}_k}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{a}_k}^{\mathrm{T}} + \boldsymbol{M}\boldsymbol{B}_k\boldsymbol{\Sigma}_x^{-1} + \boldsymbol{\Sigma}_x^{-1}\boldsymbol{B}_k\boldsymbol{M} + (\hat{a}_k^\alpha - 1)(\hat{b}_k^\alpha)^{-2}[\boldsymbol{a}_k\boldsymbol{a}_k^{\mathrm{T}} - \hat{b}_k^\alpha\boldsymbol{I}_n]$,
$\partial^2\boldsymbol{u}_i = \boldsymbol{M}\mathcal{I}_{\boldsymbol{\mu}_i}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{\mu}_i}^{\mathrm{T}} + \boldsymbol{M}\boldsymbol{B}_i\boldsymbol{\Sigma}_x^{-1} + \boldsymbol{\Sigma}_x^{-1}\boldsymbol{B}_i\boldsymbol{M}$, $\boldsymbol{B}_i = \boldsymbol{V}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{u}_i}^{\mathrm{T}}\mathcal{I}_{\boldsymbol{u}_i}\boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}$,
$\partial^2\nu_i = -\frac{N}{2}\nu_i^{-2} + \frac{1}{2}\mathrm{Tr}[\boldsymbol{B}_M\mathcal{I}_{\nu_i}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\nu_i} + \boldsymbol{\Sigma}_y^r\mathcal{I}_{\nu_i}\boldsymbol{B}_M\mathcal{I}_{\nu_i} - \boldsymbol{B}_M\mathcal{I}_{\nu_i}\boldsymbol{B}_x\mathcal{I}_{\nu_i} - \boldsymbol{B}_x\mathcal{I}_{\nu_i}\boldsymbol{B}_M\mathcal{I}_{\nu_i}] - (a_i^\nu - 1)\nu_i^{-2}$,
$\partial^2\varphi = \mathrm{Tr}[\varphi^{-3}\boldsymbol{M}] - \mathrm{Tr}[\varphi^{-4}\boldsymbol{\Sigma}_x^{-1}\boldsymbol{M}] - (a^\varphi + \frac{Nn}{2} - 1)\varphi^{-2}$.
- $\partial a_k^\alpha = \psi(\hat{a}_k^\alpha - \ln\hat{b}_k^\alpha) - 1 + \ln b_k^\alpha + (\hat{a}_k^\alpha - 1)\psi'(\hat{a}_k^\alpha) - \psi(a_k^\alpha) + \frac{1}{2}(\hat{b}_k^\alpha)^{-2}\boldsymbol{\Delta}_{\boldsymbol{a}_k}^{\mathrm{T}}(\boldsymbol{a}_k\boldsymbol{a}_k^T\mathrm{T} + \hat{b}_k^\alpha\boldsymbol{I}_n)\boldsymbol{\Delta}_{\boldsymbol{a}_k}$,
- $\partial b_k^\alpha = (\hat{a}_k^\alpha - 1)/\hat{b}_k^\alpha + a_k^\alpha - (\hat{a}_k^\alpha - 1)(\hat{b}_k^\alpha)^{-3}\boldsymbol{\Delta}_{\boldsymbol{a}_k}^{\mathrm{T}}(\boldsymbol{a}_k\boldsymbol{a}_k^{\mathrm{T}})\boldsymbol{\Delta}_{\mathbf{a}_k}$,
- $\partial a_i^\nu = \ln b_i^\nu - \psi(a_i^\nu) + \ln\nu_i - \frac{1}{2}\nu_i^{-2}(\boldsymbol{\Delta}_{\nu_i})^2$, $\partial b_i^\nu = a_i^\nu/b_i^\nu - \nu_i$,
- $\partial a^\varphi = \ln\varphi + \ln b^\varphi - \psi(a^\varphi) - \frac{1}{2}\varphi^{-2}(\boldsymbol{\Delta}_\varphi)^2$, $\partial b^\varphi = a^\varphi/b^\varphi - \varphi$.

Notations:
$\boldsymbol{M} = \boldsymbol{\Sigma}_x^{-1}\boldsymbol{S}_N\boldsymbol{\Sigma}_x^{-1}$, $\hat{a}_k^\alpha = a_k^\alpha + \frac{n}{2}$, $\hat{b}_k^\alpha = b_k^\alpha + \frac{\boldsymbol{a}_k^{\mathrm{T}}\boldsymbol{a}_k}{2}$, $\boldsymbol{B}_k = \boldsymbol{V}\boldsymbol{\Sigma}_y^r\mathcal{I}_{\boldsymbol{a}_k}^{\mathrm{T}}\mathcal{I}_{\boldsymbol{a}_k}\boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}$, $\boldsymbol{B}_w = \boldsymbol{\Sigma}_x^{-1}\boldsymbol{S}_N\boldsymbol{\Delta}_w\boldsymbol{\Sigma}_{y|x}^{-1}$,
$\mathcal{I}_{\boldsymbol{a}_k} = [\mathbf{0}_{1\times(r+k-1)}, 1, \mathbf{0}_{1\times(m-r-k)}]$, $\mathcal{I}_{\nu_i} = \mathrm{diag}[\mathbf{0}_{1\times(i-1)}, 1, \mathbf{0}_{1\times(m-i)}]$, $\boldsymbol{B}_x = \boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}\boldsymbol{\Sigma}_x^{-1}\boldsymbol{V}\boldsymbol{\Sigma}_y^r$, $\boldsymbol{B}_M = \boldsymbol{\Sigma}_y^r\boldsymbol{V}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{V}\boldsymbol{\Sigma}_y^r$,
$\mathcal{I}_{\boldsymbol{u}_i} = [\mathbf{0}_{1\times(i-1)}, 1, \mathbf{0}_{1\times(m-i)}]$, $\mathbf{0}_{\ell_1\times\ell_2}$ denotes an $\ell_1 \times \ell_2$ zero matrix.

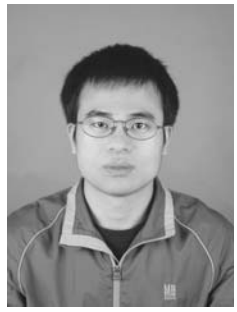
convergence: repeat until the value of the harmony functional converges.

Acknowledgements The work described in this paper was supported by a grant of the General Research Fund from the Research Grant Council of the Hong Kong SAR (No. CUHK4177/07E).

References

1. Akaike H. A new look at the statistical model identification. IEEE Transactions on Automatic Control, 1974, 19(6): 716–723
2. Schwarz G. Estimating the dimension of a model. Annals of Statistics, 1978, 6(2): 461–464
3. Rissanen J. Modelling by the shortest data description. Automatica, 1978, 14(5): 465–471
4. Anderson T, Rubin H. Statistical inference in factor analysis. In: Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability. 1956, 5: 111–150
5. Fodor I K. A survey of dimension reduction techniques. Technical Report UCRL-ID-148494. 2002
6. Jolliffe I T. Principal Component Analysis. 2nd ed. New York: Springer, 2002
7. Tipping M E, Bishop C M. Mixtures of probabilistic principal component analyzers. Neural Computation, 1999, 11(2): 443–482
8. Bishop C M. Variational principal components. In: Proceedings of the Ninth International Conference on Artificial Neural Networks. 1999, 509–514
9. Ghahramani Z, Beal M J. Variational inference for Bayesian mixtures of factor analysers. Advances in Neural Information Processing System, 2000, 12: 449–455
10. Nielsen F B. Variational approach to factor analysis and related models. Dissertation for the Master's Degree. Lyngby: Technical University of Denmark, 2004
11. Hills S E, Smith A F. Parameterization issues in Bayesian inference. Bayesian Statistics, 1992, 4: 227–246
12. Kass R E, Slate E H. Reparameterization and diagnostics of posterior nonnormality. Bayesian Statistics, 1992, 4: 289–305
13. Gelman A. Parameterization and Bayesian modeling. Journal of the American Statistical Association, 2004, 99(466): 537–545

14. Ghosh J, Dunson D B. Default prior distributions and efficient posterior computation in Bayesian factor analysis. Journal of Computational and Graphical Statistical Statistics, 2009, 18(2): 306–320
15. Xu L. Bayesian Ying Yang system and theory as a unified statistical learning approach: (i) unsupervised and semi-unsupervised learning. In: Amari S, Kassabov N, eds. Brain-like Computing and Intelligent Information Systems. Berlin: Springer-Verlag, 1997, 241–274
16. Xu L. Bayesian Ying-Yang learning theory for data dimension reduction and determination. Journal of Computational Intelligence in Finance, 1998, 6(5): 6–18
17. Xu L. BYY harmony learning, independent state space, and generalized APT financial analyses. IEEE Transactions on Neural Networks, 2001, 12(4): 822–849
18. Hu X, Xu L. A comparative investigation on subspace dimension determination. Neural Networks, 2004, 17(8–9): 1051–1059
19. Shi L, Xu L. Local factor analysis with automatic model selection: a comparative study and digits recognition application. In: Proceedings Part II of the 16th International Conference on Artificial Neural Networks. 2006, 260–269
20. Jordan M I, Ghahramani Z, Jaakkola T S, Saul L K. An introduction to variational methods for graphical models. Machine Learning, 1999, 37(2): 183–233
21. Beal M J. Variational algorithms for approximate Bayesian inference. Dissertation for the Doctoral Degree. London: University College London, 2003
22. Xu L. Bayesian Ying-Yang system, best harmony learning, and five action circling. Frontiers of Electrical and Electronic Engineering in China, 2010, 5(3): 281–328
23. Xu L. Fundamentals, challenges, and advances of statistical learning for knowledge discovery and problem solving: a BYY harmony perspective. In: Proceedings of International Conference on Neural Networks and Brain. 2005, 1: 24–55
24. Rubin D B, Thayer D T. EM algorithm for ML factor analysis. Psychometrika, 1982, 47(1): 69–76
25. Bozdogan H, Ramirez D E. FACAIC: model selection algorithm for the orthogonal factor model using AIC and FACAIC. Psychometrika, 1988, 53(3): 407–415
26. Tu S, Xu L. A study of several model selection criteria for determining the number of signals. In: Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing. 2010, 1966–1969
27. Xu L. Bayesian-Kullback coupled YING-YANG machines: unified learning and new results on vector quantization. In: Proceedings of International Conference on Neural Information Processing. 1995, 977–988
28. Xu L. Codimensional matrix pairing perspective of BYY harmony learning: hierarchy of bilinear systems, joint decomposition of data-covariance, and applications of network biology. Frontiers of Electrical and Electronic Engineering in China, 2011, 6(1): 86–119
29. Xu L. Bayesian Ying Yang learning. Scholarpedia, 2007, 2(3): 1809
30. Xu L. Bayesian Ying Yang system, best harmony learning, and Gaussian manifold based family. In: Zurada J, Yen G,Wang J, eds. Computational Intelligence: Research Frontiers. Berlin-Heidelberg: Springer-Verlag, 2008, 5050: 48–78
31. Tu S, Xu L. An investigation of several typical model selection criteria for detecting the number of signals. Frontiers of Electrical and Electronic Engineering in China, 2011 (in Press)
32. Asuncion A, Newman D. UCI machine learning repository, 2007. http://www.ics.uci.edu/ ~ mlearn/MLRepository.html



Shikui TU is a Ph.D candidate of the Department of Computer Science and Engineering, The Chinese University of Hong Kong. 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, chair professor of The Chinese University of Hong Kong (CUHK), Fellow of IEEE (2001–), Fellow of International Association for Pattern Recognition (2002–), and Academician of European Academy of Sciences (2002–). He completed his Ph.D thesis at Tsinghua University by the end of 1986, became postdoc at Peking University in 1987, then promoted to associate professor in 1988 and a professor in 1992. During 1989–1993 he was research associate and postdoc in Finland, Canada and USA, including Harvard and MIT. He joined CUHK as senior lecturer in 1993, professor in 1996, and chair professor in 2002. He published several well-cited papers on neural networks, statistical learning, and pattern recognition, e.g., his papers got over 3400 citations (SCI) and over 6300 citations by Google Scholar (GS), with the top-10 papers scored over 2100 (SCI) and 4100 (GS). One paper scored 790 (SCI) and 1351 (GS). He served as a past governor of International Neural Network Society (INNS), a past president of APNNA, and a member of Fellow Committee of IEEE CI Society. He received several national and international academic awards (e.g., 1993 National Nature Science Award, 1995 INNS Leadership Award and 2006 APNNA Outstanding Achievement Award).