Front. Electr. Electron. Eng. 2012, 7(1): 147–196
DOI 10.1007/s11460-012-0190-2

RESEARCH ARTICLE

Lei XU

On essential topics of BYY harmony learning: Current status, challenging issues, and gene analysis applications

Abstract As a supplementary of [Xu L. Front. Electr. Electron. Eng. China, 2010, 5(3): 281–328], this paper outlines current status of efforts made on Bayesian Ying-Yang (BYY) harmony learning, plus gene analysis applications. At the beginning, a bird's-eye view is provided via Gaussian mixture in comparison with typical learning algorithms and model selection criteria. Particularly, semi-supervised learning is covered simply via choosing a scalar parameter. Then, essential topics and demanding issues about BYY system design and BYY harmony learning are systematically outlined, with a modern perspective on Yin-Yang viewpoint discussed, another Yang factorization addressed, and coordinations across and within Ying-Yang summarized. The BYY system acts as a unified framework to accommodate unsupervised, supervised, and semi-supervised learning all in one formulation, while the best harmony learning provides novelty and strength to automatic model selection. Also, mathematical formulation of harmony functional has been addressed as a unified scheme for measuring the proximity to be considered in a BYY system, and used as the best choice among others. Moreover, efforts are made on a number of learning tasks, including a mode-switching factor analysis proposed as a semi-blind learning framework for several types of independent factor analysis, a hidden Markov model (HMM) gated temporal factor analysis suggested for modeling piecewise stationary temporal dependence, and a two-level hierarchical Gaussian mixture extended to cover semi-supervised learning, as well as a manifold learning modified to facilitate automatic model selection. Finally, studies are applied to the problems of gene analysis, such as genome-wide association, exome sequencing analysis, and gene transcriptional regulation.

Received November 25, 2011; accepted December 15, 2011

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 Bayesian Ying-Yang (BYY) harmony learning, harmony functional, automatic model selection, Gaussian mixture, hidden Markov model (HMM) gated temporal factor analysis, hierarchical Gaussian mixture, manifold learning, semi-supervised learning, semi-blind learning, genome-wide association, exome sequencing analysis, gene transcriptional regulation

1 Introduction

The Bayesian Ying-Yang (BYY) harmony learning is featured by seeking the best harmony between the Ying-Yang pair in a BYY system. In this system, the observation X is regarded as generated from its inner representation R. The joint distribution of X and R has two types of Bayesian decompositions. One is $p(R|X)p(X)$ called the Yang machine, while the other $q(X|R)q(R)$ is the Ying machine. The best harmony between this Ying-Yang pair is implemented via maximizing the following harmony functional:

$$H(p||q) = \int p(R|X)p(X)\ln[q(X|R)q(R)]\mathrm{d}X\mathrm{d}R, \quad (1)$$

which 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 from several perspectives as introduced in Sect. 4.1 of Ref. [1]. The spelling 'Ying' should be 'Yin' by the current Chinese Pin Yin system that could be backtracked to over 400 years evolution from the initiatives by westerns (e.g., M. Ricci, N. Trigault) who would not be aware of that the length of 'Yin' lost its harmony with Yang. In a compliment to the famous Chinese ancient harmony philosophy, 'Ying' is preferred by the present author since 1995.

Firstly proposed in Ref. [2] and systematically developed over a decade and half, the BYY harmony learning provides not only a general framework that accommodates typical learning approaches from a unified perspective but also a new road that leads to improved

model selection criteria, Ying-Yang alternative learning with automatic model selection, as well as coordinated implementation of Ying-based model selection and Yang-based learning regularization. Readers are referred to Ref. [1] for a latest systematical introduction and a tutorial on algorithms for typical learning tasks. Also, readers are referred to Ref. [3] for another perspective that a co-dimensional matrix pair forms a building unit and a hierarchy of such building units sets up the BYY system. As a supplementary of Refs. [1] and [3], this paper focuses on outlining current status of essential topics and challenging issues about the BYY harmony learning, plus new applications to gene analysis.

Taking the problem of learning Gaussian mixture to start with, Sect. 2 provides a comparative introduction on not only local learning mechanism in comparison with typical learning algorithms, such as expectation-maximization (EM), hard-cut EM, k-means, competitive learning, and rival penalized competitive learning, but also its performance of automatic model selection in comparison with Bayesian approaches and two-stage implementation based on some model selection criteria. As a supplementary to Sects. 3.1 and 2.1 in Ref. [1], where the focuses were put on basic concepts, detailed explanations, and learning algorithms, this section aims at their relations to typical learning algorithms and a systematic comparison. Also, unsupervised, supervised, and semi-supervised learning are all included in a common format simply via an option of a scalar parameter.

Reorganizing the key issues of Sect. 4.2 in Ref. [1] and also some other issues scattered among previous publications [4–6], Sect. 3 systematically outlines essential topics on designing BYY system, including basic principles, typical ingredients and their probabilistic structures. Four new issues are added. First, a modern perspective on Yin-Yang viewpoint is discussed, with further insights on the BYY system and the relation between Ying machine and Yang machine. Second, an alternative factorization of Yang machine into ingredient structures are addressed to facilitate designing Yang machine such that the integral over θ can be handled with conjugate priors [7]. Third, coordination across Ying-Yang, within Ying, and within Yang are outlined, featured not only by a coordination between Ying-based model selection and Yang-based learning regularization, but also by coordinations within ingredients of Ying and within ingredients of Yang. Fourth, the BYY system is shown to provide a unified framework to accommodate unsupervised, supervised, and semi-supervised learning all in one formulation, differing in special settings on one or more of ingredients in a BYY system. The last but not least, historical remarks are provided on past studies about BYY system, and demanding issues are also addressed.

Next, Sect. 4 outlines essential topics on the BYY harmony learning. On one hand, Sect. 4.1 and Appendix B of Ref. [1] are reorganized and elaborated systematically to show how Ying-Yang best matching provides a unified perspective that covers existing typical learning principles or approaches and how Ying-Yang best harmony not only leads to existing learning principles but also provides novelty and strength to learning with automatic model selection. On the other hand, new efforts have been made on both the harmony functional level and the learning implementation level. First, the mathematical formulation of harmony functional has been further addressed as a unified scheme for measuring bi-entity proximity, by which maximization of its special cases leads to maximizing entropy, minimizing cross entropy [8–10], and minimizing Kullback divergence (plus its related learning approaches as well). Second, how to measure bi-entity proximity in a BYY system has been examined from different perspectives. After outlining studies from a unidirectional perspective, including both typical top-down approaches [11–14] and bottom-up approaches [15–17], as well as discriminative training criteria proposed in the literature of speech recognition [18,19], we come to measuring the bi-entity proximity between Ying-Yang as the solution. Third, learning implementation has been further elaborated with detailed description on manifold shrinking dynamics and with balanced operation on handling approximation. Also, historical remarks are provided on past studies about the BYY best harmony learning theory and implementations, and challenging issues are suggested for further investigations.

Moreover, Sect. 5 provides some insights on the roles of inner dependence structures in the BYY system. First, a lattice structure based mode-switching factor analysis has been proposed as a semi-blind learning framework that summarizes Gaussian factor analysis (FA), non-Gaussian FA (NFA), and binary FA (BFA), as well as Gaussian mixture and local FA models, and accommodates unsupervised, supervised, and semi-supervised learning all in one formulation. Second, previous studies on temporal structure has been outlined, with new suggestions for modeling piecewise stationary temporal dependence (e.g., high-resolution range profile data for radar automatic target recognition) by hidden Markov model (HMM) gated temporal factor analysis and extensions. Third, we proceed to hierarchical and graphical structures, and present a two-level hierarchical Gaussian mixture, simplified from a three-level hierarchical Gaussian mixture illustrated in Fig. 12 of Ref. [1] but extended to cover semi-supervised learning. Also, we modify the manifold learning by Eq. (66) in Ref. [3] with the role of graph Laplacian matrix adjusted by learning a diagonal matrix for automatic model selection.

Finally, studies have been applied to gene analysis in Sect. 6, including genome wide association (GWA)

study, exome sequencing analysis, and gene transcriptional regulation. A semi-blind NFA learning is proposed to improve the performance of logistic regression and then extended for analyzing the relations between a set of single nucleotide polymorphisms (SNPs) and multiple complex traits. Moreover, this semi-blind NFA learning, and a three-layer network as well, is implemented by the BYY harmony learning with automatic model selection nature to push extra parameters towards zero. We observe how interactions between two SNPs affect traits or diseases, through identifying a subset of quadratic separable ability. Moreover, our efforts on SNP based analysis further proceed to exome sequencing analysis along two directions. One is getting a confusion table by one of classifiers with a good generalization ability (e.g., the above BYY harmony learning based semi-blind NFA) and testing a null hypothesis made from this confusion table with help of an appropriate statistic. The other direction is making a dimension reduction by learning a BYY system with its Yang pathway as a classifier for getting a confusion table. Furthermore, we move to modeling gene transcriptional regulation by a noisy BFA, which leads to improvements of networks component analysis [20,21]. This noisy BFA is also modified to a semi-blind BFA that can be regarded as a probabilistic extension of the semi-blind learning BFA (see Eq. (70) in Ref. [3]), for which a learning algorithm is obtained from a standard BFA algorithm with help of a three-step-alternation.

2 Begin with learning Gaussian mixture

2.1 Algorithms for learning Gaussian mixture

2.1.1 Bayes classifier by supervised learning

We start from supervised learning on Gaussian mixture as shown in Fig. 1, namely, each class is modeled by a Gaussian $G(x|\mu_\ell, \Sigma_\ell)$ with a proportion α_j. Given a set of samples with each sample x_t associated with a teaching label j or equivalently:

$$p_{\ell,t} = \delta_{\ell,j},\ \delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where $\delta_{i,j}$ is the Kronecker delta function.

The maximum likelihood (ML) estimation on parameters of each Gaussian is directly obtained as follows:

$$\alpha_\ell^* = \frac{\sum_t p_{\ell,t}}{N}, \quad \mu_\ell^* = \frac{1}{N\alpha_\ell^*}\sum\nolimits_t p_{\ell,t}x_t,$$
$$\Sigma_\ell^* = \frac{1}{N\alpha_\ell^*}\sum\nolimits_t p_{\ell,t}(x_t - \mu_\ell^*)(x_t - \mu_\ell^*)^{\mathrm{T}}, \tag{3}$$

that is, learning is decoupled into making the ML learning on each Gaussian separately. Then, a Bayes classifier for Maximum A Posteriori (MAP) classification,

$$\ell^*(x) = \arg\max_\ell q(\ell|x, \theta^*), \tag{4}$$

is simply obtained via $q(\ell|x, \theta)$ in Fig. 1 with $\theta^* = \{\alpha_\ell^*, \mu_\ell^*, \Sigma_\ell^*\}$ obtained by Eq. (3). The supervised implementation puts the computation by Eq. (3) into Eq. (4) only in one run, which is good for saving computing cost. However, the obtained label ℓ^* has not been reused to refine the original label by Eq. (2).

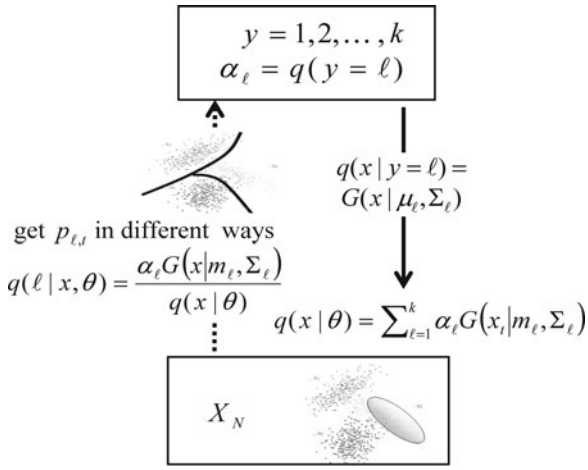

Fig. 1 Learning Gaussian mixture

2.1.2 EM algorithm and hard-cut EM

Given a set of samples without teaching labels, as shown in Fig. 1, they are collectively described by a Gaussian mixture $q(x|\theta)$. Unsupervised learning is made by using θ^* via Eq. (3) to estimate

$$p_{\ell,t} = q(\ell|x_t, \theta^*), \tag{5}$$

which is called E-step while Eq. (3) is called M-step. The E-step and M-step composes of the EM algorithm [22,23] as listed in the first row of Table 1 The two steps are iterated from an initialization of θ^*, until converged.

Samples can be classified by Eq. (4) or equivalently we get the counterpart of Eq. (2) as follows:

$$p_{\ell,t} = \delta_{\ell,\ell^*(x_t)}, \tag{6}$$

which is the winner-take-all (WTA) hard-cut modification of the E-step by Eq. (5). As listed in the second row of Table 1, making iterations by Eqs. (6) and (3) provides a hard-cut EM algorithm that was firstly suggested in Sect. 4.2 of Ref. [2]. Moreover, at the special case $\alpha_\ell = 1/k$ and $\Sigma_\ell = \sigma^2 I$, Eq. (4) is simplified into the following nearest neighbor (NN) rule:

$$\ell^*(x) = \arg\min_\ell \|x - \mu_\ell\|^2. \tag{7}$$

As listed in the third row of Table 1, the hard-cut EM algorithm is further degenerated into the conventional

Table 1 Algorithms for learning Gaussian mixture

Algorithms	Step 1	Step 2
in general	get $p_{\ell,t}$ to allocate a sample x_t among all the possible values of the inner coding y_t	update all the parameters $\theta = \{\alpha_\ell, m_\ell, \Sigma_\ell\}_{\ell=1}^k$
EM algorithm	E-step by Eq. (5)	M-step by Eq. (3)
hard-cut EM	MAP by Eqs. (4) and (6)	M-step by Eq. (3)
k-means	NN rule by Eqs. (6) and (7)	update $\mu_{\ell^*}^*$ in Eq. (3)
competitive learning	NN rule by Eqs. (6) and (7)	update $\mu_{\ell^*}^*$ in Eq. (3) or by Eq. (8)
RPCL learning	get the winner and rival by Eq. (9) based on $q(\ell \mid x_t, \theta)$	update by Eq. (3)
Ying-Yang alternation	Yang-step by Eq. (10)	Ying-step by Eq. (3)
semi-supervised BYY	Yang-step by Eq. (14)	Ying-step by Eq. (3)

k-means algorithm (or a NN-VQ algorithm) (see Sect. 4.3 of Ref. [2]).

2.1.3 Competitive learning and RPCL learning

As a sample x_t comes, we get $p_{\ell,t}$ by Eqs. (6) and (7). Then, we update

$$\mu_\ell^{\text{new}} = \mu_\ell^{\text{old}} + \eta p_{\ell,t}(x_t - \mu_\ell^{\text{old}}), \tag{8}$$

where $\eta > 0$ is a small number as a learning step size. Given a mean estimate μ_n obtained from n samples, as a new sample x_t comes we have $\mu_{n+1} = (N\mu_n + x_t)/(N+1) = \mu_n + \eta(x_t - \mu_n)$, $\eta = 1/(N+1)$. Thus, we can regard Eq. (8) as an incremental or adaptive version of updating $\mu_{\ell^*}^*$ in Eq. (3). In other words, this competitive learning is actually an incremental or adaptive version of the k-means algorithm.

RPCL learning incurs automatic model selection. In addition to that the winner μ_{ℓ^*} moves a little bit to adapt the current sample x_t, RPCL learning also repels the rival (i.e., the second winner) μ_r a little bit apart from x_t to reduce a duplicated information allocation. That is, Eq. (6) is modified into

$$p_{\ell,t} = \begin{cases} 1, & \ell = \ell^*(x_t), \\ -\gamma, & \ell = \arg\min_{\ell \neq \ell^*} \|x - \mu_\ell\|^2, \\ 0, & \text{otherwise}, \end{cases} \tag{9}$$

where $\gamma \approx 0.005 \sim 0.05$ controls the penalizing strength. The implementation of Eq. (8) will gradually drive an extra μ_j far away from data, during which the number k of clusters is determined automatically [24]. As listed in the 5th row of Table 1, Eqs. (8), (9) and (3) are generally implemented on Gaussian mixture with an extra Gaussian discarded as its corresponding $\alpha_j \to 0$ and $\text{Tr}[\Sigma_j] \to 0$. One key point of RPCL is how to control an appropriate penalizing strength γ, which is usually handled by a rule of thumb. Additionally, a direct updating of Σ_j by Eq. (3) may not guarantee that each Σ_j remains to be nonnegative definite, for which we seek some techniques (see Table 1 (B) and (C) in Ref. [25]).

2.1.4 BYY harmony learning algorithm

With $p_{\ell,t}$ given, the BYY harmony learning updates each Gaussian still by Eq. (3) as its Ying-step, namely, the Ying-step has a same formulation as the M-step, while a Yang-step modifies the E-step by Eq. (5) with one additional correcting term as follows:

$$\begin{aligned} &p_{\ell,t} = q(\ell|x_t, \theta^*)(1 + \Delta\pi_{\ell,t}), \\ &\Delta\pi_{\ell,t} = \begin{cases} \pi_t(\theta_\ell^*) - \sum_j q(j|x_t, \theta^*)\pi_t(\theta_j^*), & (a) \\ -E_t(\theta_\ell^*) + \sum_j q(j|x_t, \theta^*)E_t(\theta_\ell^*), & (b) \end{cases} \\ &\pi_t(\theta_j) = \ln[\alpha_j G(x_t|\mu_j, \Sigma_j)], \\ &E_t(\theta_j) = -\ln q(j|x_t, \theta) = -\ln \frac{\alpha_j G(x_t|\mu_j, \Sigma_j)}{q(x_t|\theta)}, \end{aligned} \tag{10}$$

where $\Delta\pi_{\ell,t}$ of choice (b) is equivalent to the choice (a). The choice (a) describes the top-down fitness on x_t by the jth Gaussian under the benchmark of the average fitness to x_t by the Gaussian mixture, while the choice (b) describes the bottom-up certainty of classifying x_t to the jth Gaussian under the benchmark of the average certainty of classifying x_t to each Gaussian.

Taking the choice (a) as an example, $\Delta\pi_{\ell,t} > 0$ means that the jth component is better than the average of all the components in term of their fitness $\pi_t(\theta_j)$. We thus update the jth component in Eq. (3) to enhance its contribution on describing x_t. If $0 > \Delta\pi_{\ell,t} > -1$, i.e., the fitness $\pi_t(\theta_j)$ by the jth component is below the average but still not too far away, the contribution of x_t on updating the jth component remains a same trend as in Eq. (3) but with a reduced strength. Moreover, when $-1 > \Delta\pi_{\ell,t}$, the updating on the jth component reverses the direction to become de-learning, somewhat similar to updating the rival in RPCL learning.

Iterating Eqs. (10) and (3) alternatively implements the maximization of the following simplified harmony functional $H(p||q)$ by Eq. (1):

$$\begin{aligned} H(p\|q) &= \sum_t \sum_j p(j|x_t, \theta)\pi_t(\theta_j), \\ p(j|x_t, \theta) &= q(j|x_t, \theta), \end{aligned} \tag{11}$$

where $\pi_t(\theta_j)$ is given by Eq. (10). Strictly speaking, putting $p_{\ell,t}$ by Eq. (10) directly into Eq. (3) may not guarantee the iteration to converge, since Eq. (3) is only an approximate solution of the fixing point equation $\nabla_\theta H(p||q) = \sum_t \sum_j p_{j,t} \nabla_\theta \pi_t(\theta_j) = 0$. However, $H(p||q)$ will increase from $\theta^{*\text{old}}$ of the current value along the direction towards the new value θ^* if it moves in an appropriate stepsize, i.e., $\theta^{*\text{old}} + \eta(\theta^* - \theta^{*\text{old}})$ (see Box ③ and Remark c in Fig. 7 of Ref. [1]), to which readers are also referred to Box ④ and Remark a) for smoothing each sample x_t by a Gaussian kernel $G(x|x_t, h^2 I)$, especially when there is only a small size of samples.

2.1.5 Semi-supervised BYY harmony learning

The BYY harmony learning is also applicable to samples with teaching labels, which leads to a classifier that is different from Bayes classifier with an improved discriminative ability. Generally, semi-supervised learning considers a mixed case of supervised learning on samples with teaching labels and unsupervised learning on samples without teaching labels. One early exploration on Gaussian mixture was made in 1997 under the name of *semi-unsupervised learning*, see Eq. (7.14) in Ref. [26]. Specifically, given a teaching pair x_t, j_t^*, the key point is that $p(j|x_t, \theta)$ in Eq. (11) is replaced by

$$p(j|x_t, \theta) = (1-\gamma) q(j|x_t, \theta) + \gamma \delta_{\ell, j_t^*}, \tag{12}$$

where $0 \leqslant \gamma \leqslant 1$ is a constant that reflects the strength of teaching. However, a weak point is that γ is irrelevant to each sample x_t.

Instead of Eq. (12), we may also consider $p(j|x_t, \theta) = q_\gamma(\ell|x_t, \theta)$ given by

$$\begin{aligned} q_\gamma(\ell|x_t, \theta) &= \frac{\gamma \delta_{\ell, j_t^*} + \alpha_\ell G(x_t|\mu_\ell, \Sigma_\ell)}{q_\gamma(x_t)}, \\ q_\gamma(x_t) &= \gamma + \sum_j \alpha_j G(x_t|\mu_j, \Sigma_j) \chi_\kappa^*(j). \end{aligned} \tag{13}$$

When $\gamma = 0$, we have $q_0(\ell|x_t, \theta) = q(\ell|x_t, \theta)$ and thus are lead to the previously introduced unsupervised learning on Gaussian mixture,while in the limit $\gamma = \infty$, we are lead to $q_\infty(\ell|x, \theta) = \delta_{\ell, j_t^*}$, which means that the teaching label is absolutely trustable and thus only supervised learning is implemented.

To implement a semi-supervised BYY harmony learning in general, the Ying-step is again updating each Gaussian still by Eq. (3), while the Yang-step gets $p_{\ell,t}$ by further modifying Eq. (10) into

$$\begin{aligned} & p_{\ell,t} = q_\gamma(\ell|x_t, \theta^*) \chi_\kappa^*(\ell) + \omega_t^\gamma q_0(\ell|x_t, \theta^*) \Delta\pi_{\ell,t}, \\ & \omega_t^\gamma = \frac{q_0(x_t)}{q_\gamma(x_t)},\ q_0(x) = q(x),\ q_0(\ell|x, \theta) = q(\ell|x, \theta), \\ & \Delta\pi_{\ell,t} \\ & = \begin{cases} \pi_t(\theta_\ell^*) \chi_\kappa^*(\ell) - \sum_j q_\gamma(j|x_t, \theta^*) \pi_t(\theta_j^*) \chi_\kappa^*(j), & 1) \\ -E_t(\theta_\ell^*) \chi_\kappa^*(\ell) + \sum_j q_\gamma(j|x_t, \theta^*) E_t(\theta_j^*) \chi_\kappa^*(j), & 2) \end{cases} \\ & j_t^* = \begin{cases} j_{t,1}^*, & \text{unsupervised with no label for } x_t, \\ j_t, & \text{for each supervised pair } x_t, j_t, \end{cases} \\ & \chi_\kappa^*(\ell) = \begin{cases} 1, & \text{for } \ell \in J_t^\kappa, \\ 0, & \text{for } \ell \notin J_t^\kappa, \end{cases} \\ & J_t^\kappa = \{j_t^*\} \cup \{j_{t,1}^*, j_{t,2}^*, \ldots, j_{t,\kappa}^*\} \subseteq \{1, 2, \ldots, k\}, \\ & \text{with } \pi_t(\theta_{j_{t,1}^*}^*) \geqslant \cdots \geqslant \pi_t(\theta_{j_{t,\kappa}^*}^*) \cdots \geqslant \pi_t(\theta_{j_{t,k}^*}^*), \end{aligned} \tag{14}$$

where the indices are sorted by the values of π_t, and a subset with the largest values is selected as J_t^κ.

When $\gamma = 0$, we start from $J_t^\kappa = \{1, 2, \ldots, k\}$ at which Eq. (14) degenerates back to Eq. (10). That is, it returns to the BYY harmony learning on samples without teaching labels. Taking the choice 1) in Eq. (14) as an example, we have

$$p_{\ell,t} = q(\ell|x_t, \theta^*)(\chi_\kappa^*(\ell) + \Delta\pi_{\ell,t}), \tag{15}$$

which allocates x_t among a more concentrated subset J_t^κ. That is, we are lead to an apex approximation based BYY harmony learning on samples without teaching labels, which is helpful when k is a large number. Details are referred to Sect. 4.3 of Ref. [1].

We further have $\chi_\kappa^*(\ell) = \delta_{\ell, j_t^*}$ when J_t^κ consists of merely j_t^*, which is also reached by $p_{\ell,t}(\theta)$ in Eq. (59) of Ref. [3] as $J_t = \{j_t^*\}$ and $\Psi^x = 0$. That is, this supervised learning case is same as the one by Eq. (59) of Ref. [3].

When $\gamma \neq 0$, still taking the choice 1) in Eq. (14) as an example, we observe

$$\begin{aligned} p_{\ell,t} &= q_\gamma(j_t^*|x_t, \theta^*) \delta_{\ell, j_t^*} + \omega_t^\gamma q(\ell|x_t, \theta^*) \Delta\pi_{\ell,t}, \\ \Delta\pi_{\ell,t} &= [\delta_{\ell, j_t^*} - q_\gamma(j_t^*|x_t, \theta^*)] \pi_t(\theta_{j_t^*}^*), \end{aligned} \tag{16}$$

by which $\Delta\pi_{\ell,t} > 0$ at $\ell = j_t^*$ enhances the learning on $G\left(x_t \left| \mu_{j_t^*}, \Sigma_{j_t^*}\right.\right) \alpha_{j_t^*}$ for describing x_t, while $\Delta\pi_{\ell,t} < 0$ makes each of the rest Gaussians delearning, with a penalized strength proportional to $\omega_t^\gamma q_\gamma(j_t^*|x_t, \theta^*) q(\ell|x_t, \theta^*)$.

We have $0 \leqslant 1 - \omega_t^\gamma \leqslant 1$ that also reflects the degree of teaching when j_t^* is a teaching label or a degree of boosting when $j_t^* = \arg\max_\ell q(\ell|x_t, \theta^*)$. Specifically, $1 - \omega_t^\gamma$ increases monotonically as $\gamma > 0$ increases towards ∞, and $\gamma \delta_{\ell, j_t^*}$ makes a super-Bayes (to be further discussed later around Eq. (46) and Item (A) in Sect. 3.4.2) enhancement on j_t^*. In the limit $\gamma = \infty$ and thus $\omega_t^\gamma = 0$, it means that merely the teaching label is considered or the WTA learning is adopted. Typically, γ could be set at a large enough constant or initialized large enough and then reduces gradually during learning. Also, we may learn an appropriate γ with help of a prior $q(\gamma)$ of Gamma distribution. Therefore, we observe that unsupervised learning, supervised learning, and semi-supervised learning are considered in a common format simply via a option of γ, featured with a correcting term $\Delta\pi_{\ell,t}$ that makes each class $G\left(x_t \left| \mu_j, \Sigma_j\right.\right) \alpha_j$ become more discriminative from each other.

Actually, the alternative implementation of Eqs. (14) and (3) can be regarded as a simplification of Eqs. (56) and (57) in Ref. [3], after ignoring a priori and shutting off the de-noise nature by letting $\Psi = 0$. On the other hand, it extends the one in Ref. [3] with one teaching degree ω_t^γ added in consideration.

Conventionally, supervised learning gets a classifier $\ell^*(x)$ by Eqs. (2) and (4) from training samples with teaching labels. Then, the obtained classifier is applied to testing samples without teaching labels, while testing samples has no contribution on learning the classifier.

For many applications, especially in bioinformatics (e.g., noncoding RNA analysis), we usually have a small size of training samples and naturally expect to use testing samples to help learning the classifier as well. For this purpose, we suggest to put both the training and testing samples altogether under the above formulation of semi-supervised learning with help of the algorithm of Ying-Yang alternation listed in Table 1, while teaching labels take their roles via γ. During learning, each sample is classified by j_t^*. In addition to a better use of samples, this learning also takes the advantage of automatic model selection on determining the number of classes. Even when the number of classes is known for a set of training samples, putting both the training and testing samples together facilitates that classes may need to re-adjust or new classes may emerge.

2.2 Automatic model selection, prior aided learning, and model selection criteria

2.2.1 Model selection: Two-stage enumeration and stepwise searching

Learning a Gaussian mixture includes parameter learning for estimating all the unknown parameters θ and model selection for determining the number k of Gaussian components. Parameter learning may be made by any one of the algorithms introduced previously, while model selection for k can be made by RPCL learning and BYY harmony learning.

The EM algorithm implements the ML learning. However, maximizing the likelihood function or minimizing a fitting error suffers an over-fitting problem. More specifically, it is under-selective on k such that extra resource of structures is wasted on fitting noises or outliers. Even worse, it deteriorates the generalization performance.

Selecting k is typically handled by a two-stage learning implementation as follows:

Stage I : enumerates k to get a set M of candidate models with the parameters of each candidate estimated by the EM algorithm;
Stage II: one best candidate is selected by a model selection criterion $k^* = \arg\min_k J(k)$. (17)

Examples of such criteria include AIC, CAIC, BIC/MDL, etc. However, this two-stage implementation suffers from a huge computation because it requires parameter learning for each $k \in M$. Moreover, a larger k often implies more unknown parameters, which makes parameter estimation become less reliable and thus the criterion evaluation reduce its accuracy (see Sect. 2.1 in Ref. [1] for a detailed discussion).

One road to reduce computing cost is a stepwise implementation. Typical examples are those incremental algorithms that attempt to incorporate as much as possible what already learned as k increases step by step, focusing on learning newly added parameters [27]. Usually, it leads to a suboptimal performance. Reversely, this problem may be lessened in a way that k decreases step by step in a tree searching, for which a depth-first searching suffers from a suboptimal performance seriously, while a breadth-first searching suffers a huge combinatorial computing cost.

2.2.2 Automatic model selection: From RPCL to BYY harmony learning

An alternative road of studies is referred as automatic model selection that automatically determines k during parameter learning. Being a quite difference nature from a usual stepwise implementation that adds or removes a subset of parameters from θ based on whether a selection criterion indicates an improvement, automatic model selection is associated with a learning algorithm or a learning principle with the following two features:

- There is an indicator on a subset $\theta_{\rm SR}$ of scale representative (SR) parameters, representing a particular structural component that is effectively discarded if its corresponding $\psi(\theta_{\rm SR}) = 0$, e.g., a Gaussian component in Fig. 1 is discarded if its corresponding $\alpha_l = 0$.
- During implementation of this learning, there is an intrinsic mechanism that drives $\psi(\theta_{\rm SR}) \to 0$, if the corresponding structure is redundant and thus can be effectively discarded.

Readers are referred to page 287 of Ref. [1] for further details about automatic model selection.

One early effort is RPCL learning, with its penalizing strength handled by a rule of thumb. Favorably, the BYY harmony learning gets rid of the difficulty of finding an appropriate penalizing strength, with both parameter learning and model selection made under the best harmony principle by Eq. (1). With help of the least complexity nature, the BYY harmony learning is capable of automatic model selection even without imposing

a priori on the parameters, e.g., merely via $\Delta\pi_{\ell,t}$ in Eqs. (10), (14), (15), and (16).

2.2.3 Prior aided automatic model selection

As addressed at the end of Sect. 4.2 in Ref. [1], some prior actually intends to cancel out certain bias implicitly incurred from using a parametric model on a small size of samples, which leads to a learning regularization that effectively reduces model complexity without necessarily discarding extra parameters. On the other hand, sparse learning or Lasso shrinkage prunes away extra weights by a Laplace prior in a regression task [28,29]. Also, with the help of appropriate priors, efforts have been made on minimum message length (MML) [30] and variational Bayes (VB) for Gaussian mixture by pruning either or both of extra α_l and extra Σ_l in Refs. [13,14,31], sharing the features of automatic model selection.

Minimizing a two-part message for a statement of model and a statement of data encoded by that model, MML involves a term that computes the determinant of Fisher information matrix [32]. Figueiredo and Jain in Ref. [30] developed such an MML algorithm for learning Gaussian mixture with a prior by a product of independent Jeffreys priors on α_l and the parameters in Gaussian components, and with the Fisher information matrix approximated by a block-diagonal matrix. Tackling the difficulty in computing the marginal likelihood with a lower bound by means of variational method, the existing VB algorithms for learning Gaussian mixture are featured by a Dirichlet prior on α_l and an *independent* Normal-Wishart (NW) prior on each Gaussian component's parameters [13,31].

However, these efforts highly depends on choosing an appropriate prior, which is usually a difficult task, while an inappropriate prior may deteriorate the performance of model selection seriously. Without any priors on the parameters, VB and MML all degenerate to the maximum likelihood learning, while the BYY harmony learning is still capable of automatic model selection. Also, the performances of BYY harmony learning can be further improved by incorporating appropriate priors.

2.2.4 Empirical comparative investigation

Recently in Ref. [7], an empirical comparative investigation has been made on VB, MML, and BYY harmony learning, for learning Gaussian mixture with an appropriate number of Gaussian components determined automatically during learning. The performances are evaluated through extensive experiments with help of not only Jeffreys priors but also conjugate Dirichlet-Normal-Wishart (DNW) priors, resulting in several empirical findings.

First, as Jeffreys prior is replaced by the DNW prior, all the three approaches improve their performances. Moreover, Jeffreys makes MML slightly better than VB, while the DNW makes VB better than MML. Second, as the hyper-parameters of DNW prior are further optimized by each of its own learning principle, BYY improves its performances while VB and MML deteriorate their performances when there are too many free hyper-parameters. Actually, VB and MML lack a good guide for optimizing the hyper-parameters of DNW prior. Third, BYY considerably outperforms both VB and MML for any type of priors and whether or not hyper-parameters are optimized. Being different from VB and MML that rely on appropriate priors to perform model selection, BYY does not highly depend on the type of priors. It has model selection ability even without priors and performs already very well with Jeffreys prior, and incrementally improves as Jeffreys prior is replaced by the DNW prior.

Finally, all the algorithms were applied to image segmentation on the Berkeley database of real world images. Again, BYY outperforms VB and MML considerably, with a better ability to detect the objects that are even highly confused with the background.

2.2.5 Model selection criteria

Though a two-stage implementation by Eq. (17) suffers from a huge computation, extra computing cost will indeed bring a further improvement on the performance of model selection since $H(p||q)$ by Eq. (1), as well as the objective functions that contains some not differentiable terms about k. These terms can not be used to guide parameter learning, but is still useful for making model selection by $k^* = \arg\min_k J(k)$.

Using a prior $q(\theta|\Xi)$ with hyper-parameters Ξ, we maximize $H(p||q)$ by Eq. (1) with respect to $R = \{Y, \theta, \Xi, k\}$, by alternating the following multiple stage:

$$\begin{aligned}
&\text{Step } Y: Y^* = \arg\max_Y H(p\|q)_{\text{given } \theta,k,\Xi},\\
&\text{Step } \theta: \theta^* = \arg\max_\theta H(p\|q)_{\text{given } Y,k,\Xi},\\
&\text{Step } \Xi: \Xi^* = \arg\max_\Xi H(p\|q)_{\text{given } Y,\theta,k},\\
&\text{Step } k: k^* = \arg\min_k J(k),\\
&\qquad\qquad \text{where } J(k) = -H(p\|q)_{\text{given } Y,\theta,\Xi}, \quad (18)
\end{aligned}$$

for which readers are further referred to Sect. 4.3.

For a Gaussian mixture, Step Y gets $p_{\ell,t}$ by either of Eqs. (10), (14), (15), and (16), while Step θ becomes the one by Eq. (3). Two steps jointly correspond to Stage I in Eq. (17), while Step k corresponds to Stage II. Moreover, it follows from Eq. (13) in Ref. [1] that we have

$$J(k) = 0.5\sum_{j=1}^{k}\alpha_j\left\{\ln|\Sigma_j| + h^2\text{Tr}\left[\Sigma_j^{-1}\right]\right\}$$

$$-\sum_{j=1}^{k}\alpha_j \ln \alpha_j + 0.5 n_f(\theta), \quad (19)$$

where $n_f(\theta)$ is the number of free parameters in θ, e.g., $n_f(\theta) = dk + k - 1 + 0.5d(d+1)k$. The data smoothing parameter h could be obtained by Box ④ in Fig. 7 of Ref. [1] or by Eq. (60) given at the end of Sect. 4.3.1 in this paper. For simplicity, we may simply set $h = 0$.

Favorably, the BYY harmony learning is also able to learn hyper-parameters Ξ by Step Ξ, which may be omitted when Ξ is pre-specified or a prior $q(\theta)$ is used without hyper-parameters.

3 Topics on Bayesian Ying-Yang system

3.1 Bayesian Ying-Yang system from a modern Yin-Yang perspective

3.1.1 A modern perspective on Yin-Yang viewpoint

We refer the modern perspective on Yin-Yang introduced in Appendix B of Ref. [1] and especially Fig. B1. For convenience, we simplify Fig. B1 there into Fig. 2 here for a further insight on a Ying-Yang system. From the view of the Chinese ancient Yin-Yang (or preferably Ying-Yang) theory, a body or a system that survives and interacts with its world can be regarded as a Ying-Yang system that functionally composes of two complement parts. One is called Yang, from its external world into its inside, e.g., the bottom-up part on the left side of Fig. 1, while the other is called Ying, from its inside into its external world, e.g., the top-down part on the right side of Fig. 1,

It follows from Figs. 2(b) and 2(c) that Ying is primary, supports Yang, and demonstrates itself via Yang; e.g., Ying in Fig. 1 describes observation by a mixture of Gaussian components, which provides a basis for Yang to solve classification problem via Bayes posteriori. The performance of Yang demonstrates the goodness of Ying modeling. On the other hand, Yang is secondary, basing on Ying and serving Ying, e.g., Yang in Fig. 1 is either directly the Bayes inverse by Eq. (5) of a Gaussian mixture modeled by Ying, or a variants or extension by one of Eqs. (6), (9), (10), (15), and (16) that relate to this Bayes inverse.

Formally, as shown in Fig. 2(a), Yang consists of a visible domain X (called Yang domain) that collects samples from the external world, and a pathway $X \to R$ (called Yang pathway) that transforms what gathered in the Yang domain into inner representations to be supplied to the Ying domain R. In Fig. 1, a set X_N of samples is considered in the Yang domain. The Yang pathway consists of an estimation $X_N \to \theta^*$ by Eq. (10) and a mapping of sample x_t into either $\ell^*(x_t)$ by Eq. (6) or generally $p_{\ell,t}$ by one of Eqs. (9), (10), (15), and (16). Moreover, automatic model selection determines an appropriate mapping $X_N \to k^*$.

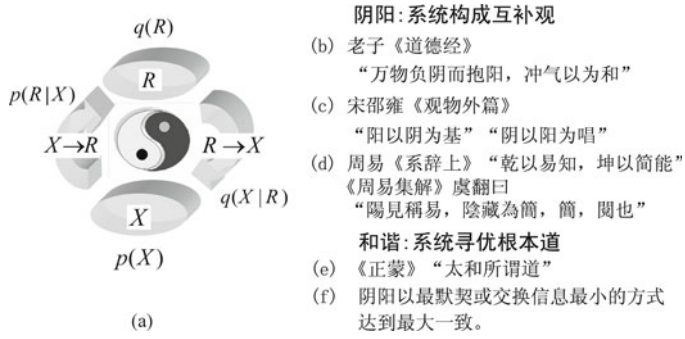

Fig. 2 BYY system from a perspective on Yin-Yang

On the other hand, Ying consists of an inner domain (called Ying domain) that receives, accumulates, integrates, digests the inner representations provided by Yang, and of a pathway $R \to X$ (called Ying pathway) that selects among the Ying domain to generate outputs to its external world. In Fig. 1, the Ying domain is simply $R = \{Y, \theta, k\}$ with $\theta = \{\alpha_j, \mu_j, \Sigma_j\}_{j=1}^k$ or $R = \{Y, \theta, k, \Xi\}$ if a priori is considered on θ with hyper-parameter Ξ. Moreover, the Ying pathway $R \to X$ is simply given by each Gaussian $G(x|\mu_j, \Sigma_j)$, while a best value of $\{\theta^*, k^*, \Xi^*\}$ is determined via learning, which specifies a Ying pathway $Y_N \to X_N$ given θ^*, k^*, Ξ^*.

As outlined in Fig. 3, the rest of this section addresses essential topics on Bayesian Ying-Yang system and its design. Continuing the above introduction from a modern Yin-Yang perspective, we further describe a Ying-Yang system in term of probabilistic modeling, namely, two Bayesian decompositions of the joint distribution on X and R, as introduced at the beginning of Sect. 1. Provided with a set X_N of samples, we need not only to determine all unknowns in R, but also to design the probabilistic structures of $p(R|X)p(X)$ and $q(X|R)q(R)$.

3.1.2 Ying machine versus Yang machine

First, Ying is primary, and its probabilistic structure is designed according to the nature of learning tasks. It follows from Fig. 2(d) that Ying is featured with a compact capacity of accommodating and accumulating, as well as a good ability of integrating and digesting whatever supplied by Yang. Thus, the probabilistic structure of $q(X|R)q(R)$ should be as compact as possible, subject to a principle of least complexity. Second, Yang is secondary and its probabilistic structure is designed according to the probabilistic structure of Ying. Moreover, it follows from Fig. 2(d) that Yang is vigor in adapting to not only variety of external world but also serving the

demands of Ying in a range of variety. Accordingly, $p(R|X)$ is designed as some type of inverse of $q(X|R)q(R)$ to match the needs of Ying, subject to a variety preservation principle.

We take Fig. 1 as an example to get some insights. Typically, $p(X)$ comes directly from X_N by the empirical distribution $p(X) = \delta(X - X_N)$, where $\delta(u)$ is the Dirac delta function. When there is a small size of samples, x_t is not directly input but smoothed by a Gaussian kernel $G(x|x_t, h^2 I)$ featured with a bandwidth h, e.g., $p(X)$ may be given by either of two Parzen window estimators in Eq. (6) of Ref. [1]. Concisely, we may regard h also as a type of data with a density $p(h|h_0)$. Jointly, we consider $p(X)$ in a general formulation as follows:

$$\begin{aligned} p_h(X|X_N) &= p(X|X_N, h)p(h|h_0), \\ p(X|X_N, h) &= \begin{cases} \prod_{t=1}^N G(x|x_t, h^2 I), & \text{(a)} \\ \prod_{t=1}^N p_h(x_t), & \text{(b)} \end{cases} \\ p_h(x) &= \frac{1}{N}\sum_{t=1}^N G\left(x\,|x_t, h^2 I\right). \end{aligned} \tag{20}$$

The structure of Ying machine $q(X|R)q(R)$ is designed subject to a least complexity principle, for which $q(R)$ is designed for a representation R of a least redundancy. In the cases without any given knowledge, a least redundancy is featured by the following independence:

$$\begin{aligned} &q(R) = q(Y)q(\theta), \\ &q(Y) \text{ in its representation of least redundancy,} \\ &q(\theta) \text{ in its representation of least redundancy.} \end{aligned} \tag{21}$$

In Fig. 1, $q(Y)$ is simply $q(y = \ell) = \alpha_\ell$, $\ell = 1, 2, \ldots, k$, for which a least redundancy means that k takes a value as small as possible. Also, we expect that the corresponding top-down $q(x|\theta)$ describes X_N well, which however tends to use a large k. Such a trade-off makes the value k unable to be determined simply by design. Instead, an appropriate k should be determined via learning, as to be further discussed later in Sect. 4.2.

For $q(\theta)$, a least redundant representation is given by

$$q(\theta) = q(\alpha)\prod_j q(\mu_j)\prod_j q(\Sigma_j), \tag{22}$$

where $\alpha = \{\alpha_\ell,\ \ell = 1, 2, \ldots, k\}$. A simple choice is that each of $q(\alpha)$, $q(\mu_j)$, and $q(\Sigma_j)$ is given by Jeffreys prior, while a more sophisticated choice is that $q(\alpha)$, $q(\mu_j)$, and $q(\Sigma_j)$ are given by the DNW prior. Readers are referred to Ref. [7] for a recent systematic study, and also to the next subsection for further discussions on $q(\theta)$.

We design the structure of $q(X|R)$ based on the numeric types of Y and X as well as their relation $Y \to X$, subject to still a principle of least complexity. In Fig. 1, $q(X|R)$ is simply a Gaussian $G(x|\mu_j, \Sigma_j)$, for which there is nothing that could be done by design to make $G(x|\mu_j, \Sigma_j)$ to be a least complexity. Even so, we still prefer to design $q(X|R)$ such that it makes a least complexity be easier reachable.

A typical designing principle for $q(X|R)$ is *divide and conquer*. That is, dividing $q(X|R)$ into a series of simple components and then combining them in a simple rule or equivalently letting $q(X|R)$ to be formed from a series of simple components in a simple composing way. E.g., considering $q(X|R) = G(x|\mu_j, \Sigma_j)$ with Σ_j in an eigen-composition $\Sigma_j = \sigma_j^2 I + \sum_{i=1}^{m_j} \lambda_{ji}\phi_{ji}\phi_{ji}^{\mathrm{T}}$, where λ_{ji}, $i = 1, 2, \ldots, m_j$ are the first m_j largest eigenvalues of Σ_j such that $\Sigma_j\phi_{ji} = \lambda_{ji}\phi_{ji}$. Such a structure makes it easier to consider $q(X|R)$ in a least complexity via determining a smallest dimension m_j.

Next, we design each of two components in the following factorization:

$$p(R|X) = p(Y, \theta|X) = p(Y|X, \theta)p(\theta|X). \tag{23}$$

In Fig. 1, we further have $p(Y|X, \theta) = \prod_t p(\ell_t|x_t, \theta)$, with the structure of $p(\ell|x, \theta)$ being the Bayes inverse $q(\ell|x, \theta)$. This type of Yang preserves equally the Ying variety about ℓ by conditioning on x, θ, which is different from the EM algorithm with $q(\ell|x, \theta^*)$ by Eq. (5) in that here we have $p(\ell|x, \theta) = q(\ell|x, \theta)$ for any θ instead of being fixed merely at θ^*. It is this difference that leads to $p_{\ell,t}$ given by Eq. (10) instead of by Eq. (5).

Conceptually, $p(\theta|X)$ may also be a Bayes inverse of its counterpart in the Ying machine. Usually, such a $p(\theta|X)$ is computationally not trackable due to computing the integral over θ. Previously in Sect. 2, we only consider one rough approximation simply by

$$p(\theta|X) = \delta(\theta - \theta^*), \tag{24}$$

where θ^* is the latest available value about θ, e.g., by Eq. (18) or Eq. (3). Readers are referred to Sect. 4.3.1 for another approximation.

3.2 Topics on BYY system design

3.2.1 Topics on Ying structure design

We start from the design of Ying structure as outlined by the Box ② in Fig. 3. The first topic considers how to describe the regularity underlying a given set X_N of samples by a combination of $q(X|R)$ for a mapping $Y \to X$ and $q(Y)$ for the representation of X_N. Then, it follows the second topic of designing appropriate structure for each components of $q(\theta)$ according to the roles of parameters in $q(X|R)$ and $q(Y)$.

The first topic is featured by different choices that trade off the roles between $q(Y)$ and $q(X|R)$. Taking Fig. 1 as an example, Y consists of only one integer ℓ taking several values and thus $q(Y)$ is simply a discrete point distribution on these values, with each value of ℓ allocating one Gaussian $G(x|\mu_j, \Sigma_j)$ to a cloud of data.

One extreme is that ℓ only takes one value, at which $q(Y)$ actually disappears and we use only one Gaussian or a general distribution to describe X_N.

The other extreme is that $q(X|R)$ degenerates into $X = Y + E$ with a noise E independent from Y that is a de-noising representation of X, for which a complicated $q(Y)$ is needed to describe the structure underlying X_N.

Between the two extremes, one typical scenario is considering that $X_N = [x_1, x_2, \ldots, x_N]$ is a $d \times N$ matrix with each column vector $x_t = [x_t^{(1)}, x_t^{(2)}, \ldots, x_t^{(d)}]^{\mathrm{T}}$ independently and identically distributed (i.i.d.). In such a case, the problem is simplified to describe x_t by its inner encoding y_t through a mapping $y_t \to x_t$. Without any given knowledge, $y_t = [y_t^{(1)}, y_t^{(2)}, \ldots, y_t^{(m)}]^{\mathrm{T}}$ with $m \geqslant 1$ is usually assumed to be mutually independent among its elements for a least redundant representation, while the mapping $y_t \to x_t$ captures the inter-dimensional dependence among the elements of $x_t = [x_t^{(1)}, x_t^{(2)}, \ldots, x_t^{(d)}]^{\mathrm{T}}$. Typically, the linear dependence is considered by

$$\begin{aligned} &x_t = Ay_t + e_t,\ Ee_t y_t^{\mathrm{T}} = 0, \\ &e_t \sim G(e_t|0, \sigma^2 I), \end{aligned} \tag{25}$$

which has been widely studied under the name of independent factor analysis, e.g., factor analysis (FA) when

$$y_t \sim G(y_t|0, \Lambda), \quad \Lambda \text{ is diagonal}, \tag{26}$$

which is different from the traditional FA that uses the following parameterization:

$$y_t \sim G(y_t|0, I). \tag{27}$$

Shortly, we use FA-a to denote FA by Eqs. (25) and (27) that jointly considers a Gaussian $G(x|\mu, \Sigma)$ via $\Sigma = \sigma^2 I + AA^{\mathrm{T}}$, while FA-b denotes FA by Eqs. (25) and (26) that jointly considers a Gaussian $G(x|\mu, \Sigma)$ via $\Sigma = \sigma^2 I + A\Lambda A^{\mathrm{T}}$. Either of two FA types considers a Gaussian $G(x|\mu, \Sigma)$ with Σ formed from several simple units in a simple composition. Each unit is simply a subspace spanned by one column vector of A, plus an additional dimension $\sigma^2 I$ for the noise e. It follows from the least complexity principle that the number of such units (or the dimension m of y_t) is as less as possible, which correspondingly is the counterpart of selecting k in Fig. 1, i.e., the task of model selection.

Two FA types are equivalent in term of maximizing the likelihood on $\ln G(x|\mu, \Sigma)$ as long as $A_a A_a^{\mathrm{T}} = A_b \Lambda A_b^{\mathrm{T}}$ or $A_a = A_b \Lambda^{0.5}$, where A_a and A_b correspond to A in FA-a and FA-b, respectively. However, the BYY harmony learning produced different model selection performances on two FA types (see Item 9.4 in Ref. [26] and Sect. 3 in Ref. [33]). Extensive empirical experiments in Ref. [34] has further shown that the BYY harmony learning and VB perform reliably and robustly better on FA-b than on FA-a, while BYY further outperforms VB considerably, especially on FA-b. In the sequel, we consider merely this type FA-b (the subscript b is omitted whenever there is no confusion) and its extensions.

Moreover, we proceed to NFA when $y_t^{(j)}$ is non-Gaussian, and BFA when $y_t^{(j)}$ takes only a binary value.

Alternatively, we may also proceed to a mixture of several FA models. That is, Eq. (25) becomes

$$\begin{aligned} &x_t = Ay_{t,\ell} + \mu_\ell + e_{t,\ell},\ Ee_{t,\ell} y_{t,\ell}^{\mathrm{T}} = 0, \\ &q(\ell) = \alpha_\ell,\ \ell = 1, 2, \ldots, k, \\ &e_{t,\ell} \sim G(e|0, \sigma_\ell^2 I), \end{aligned} \tag{28}$$

and Eq. (26) correspondingly becomes

$$y_{t,\ell} \sim G(y|0, \Lambda_\ell), \quad \Lambda_\ell \text{ is diagonal}, \tag{29}$$

which has been widely studied under the name of local factor analysis (LFA) with each located at μ_ℓ [35], see a recent review in Ref. [17]. In this case, y_t comes from

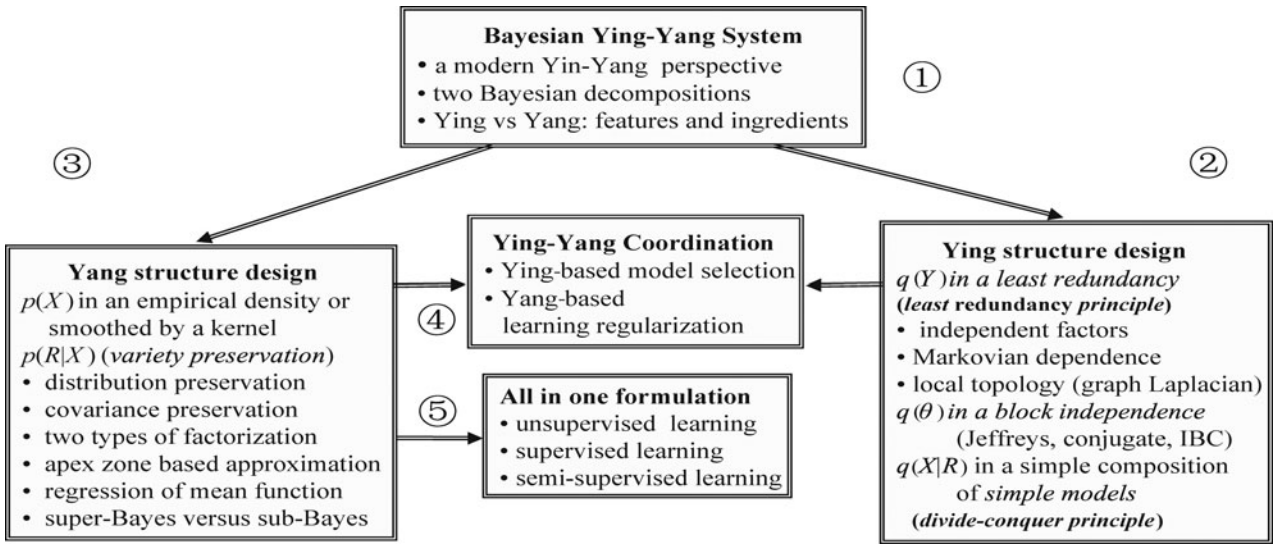

Fig. 3 BYY system design

$G(y|0,\Lambda_\ell)$ with a probability α_ℓ, and the mapping $y_t \to x_t$ is replaced by $\ell_t, y_t \to x_t$ that makes $y_t \to x_t$ locally at ℓ_t.

In parallel to these LFA studies, there are also related studies called a mixture of factor analysis [27,36,37], which actually considers a mixture of FA-a models. Thus, its difference from the above LFA is similar to the above discussed difference of FA-a from FA-b.

For recent outlines of studies on local factor analysis and independent factor analysis (FA, NFA, BFA, etc.), readers are referred to Sect. 3.2 in Ref. [1] (see the first column of its roadmap in Fig. 3) and to Ref. [3] under the guidance of its roadmap in Fig. 1. Moreover, extensions along this direction will be further discussed in Sect. 5 of this paper.

When there is some dependence structure across the columns of X_N, which is not able to be captured by Eq. (25), the Ying machine accordingly needs a structure to capture this dependence too.

There could be different types of dependence across N columns of X_N. One most commonly encountered one is serial or temporal dependence among $x_t,\ t = 1, 2, \ldots, N$. Typically, we describe this temporal dependence independently by each element of y_t, e.g., by the following first order independent autoregressive process:

$$\begin{aligned} &y_t = By_{t-1} + \varepsilon_t,\ Ey_{t-1}\varepsilon_t^{\mathrm{T}} = 0, \\ &\varepsilon_t \sim G(\varepsilon_t|0,\Lambda),\quad \Lambda \text{ is diagonal}, \\ &B = \mathrm{diag}[b_1, b_2, \ldots, b_m], \end{aligned} \tag{30}$$

while $q(X|R)$ can be simply described by the instantaneous linear relation by Eq. (25). Though a same type of temporal dependence can be described by $x_t = Cy_{t-1} + A\varepsilon_t + e_t$ with a constraint $C = AB$, it almost doubles the number of parameters in consideration, which not only makes the problem of a small sample size become more serious, but also increases the complexity of R. Thus, it follows from the least complexity principle that the combination of Eqs. (25) and (30) is preferred.

There are three typical ways for $q(Y)$ to model temporal dependence as follows:

- Transfer distribution: When $y_t = \ell_t$ is simply one integer as in Fig. 1, a transfer distribution $q(y_t = \ell_t|y_{t-1} = \ell_{t-1})$ is used to capture the first order dependence, which leads to HMM model and variants (see Sect. 5.3 in Ref. [1]).
- Mean regression: When y_t is a real vector, Eq. (30) is one example of the mean vector regression. Also, such a regression may be modified to a binary vector y_t with help of a post-linear function, e.g., see Eq. (104) in Sect. 5.5.1. Details are referred to Sect. 5.3 in Ref. [1] and Sect. 4.2 in Ref. [3].
- Covariance regression: Actually, Eq. (30) also provides a covariance matrix regression $E(y_t y_t)^{\mathrm{T}} = BE(y_{t-1}y_{t-1}^{\mathrm{T}})B^{\mathrm{T}} + \Lambda$ (see Fig. 8 in Ref. [38]).

Another type of dependence is local topology described by a nearest neighbor graph, which will be further introduced later at the end of Sect. 5.2.

After the structures of $q(X|R)$ and $q(Y)$ have been designed, there are still a set θ of unknown parameters in these structures, for which we need an appropriate prior $q(\theta)$, as encountered in the Bayesian approaches, from which those existing ways of getting $q(\theta)$ can be directly adopted, e.g., as previously discussed around Eq. (22).

Moreover, we believe that in a given parametric model $q(x|\theta)$ there is another role for a prior $q(\theta)$ to take. For a finite set of samples $\{u_t\}$, we have

$$\int q(u|\theta)\mathrm{d}u = 1,\ \text{but}\ Z(\theta) = \sum_t q(u_t|\theta) \neq 1, \tag{31}$$

which actually imposes an implicit measure or a priori on θ that incurs some unexpected bias on estimating θ. Instead of imposing a priori, another role of $q(\theta)$ should be removing this unwanted bias, which motivated the so-called induced bias cancellation (IBC) that uses $q(u_t|\theta)/Z(\theta)$ in place of $q(u_t|\theta)$, e.g., see Eq. (21) in Ref. [39] and Sect. II(A) of Ref. [40]. Readers are further referred to Sect. 3.4.3 in Ref. [41] for a recent overview and to Sect. 23.7.4 in Ref. [42] for historical remarks. Also, as pointed out at the end of Sect. 4.2 of Ref. [1], it coincides with the normalized maximum likelihood (NML) used in the MDL encoding [43].

In summary, a Ying machine is designed according to a least complexity principle, featured with designing $q(R)$ in a least redundancy principle and designing $q(X|R)$ in a divide-conquer principle. Design starts from a trade off consideration between $q(Y)$ and $q(X|R)$. Each component of $q(\theta)$ is designed according to the roles of parameters in $q(X|R)$ and $q(Y)$.

3.2.2 Topics on Yang structure design

We proceed to the Box ③ in Fig. 3. With $p(X)$ given by either $\delta(X - X_N)$ or Parzen estimators in Eq. (6) of Ref. [1], the task is mainly designing the probabilistic structure of $p(R|X)$.

For $p(R|X)$ in Eq. (23), we need to design the probabilistic structure of each component. As introduced at the end of Sect. 3.1.2, it follows from $p(Y|X,\theta) = \prod_t p(\ell_t|x_t,\theta)$ that the Bayes inverse type of Yang $p(\ell|x,\theta) = q_{\mathrm{bayes}}(\ell|x,\theta)$ serves Ying in an equal variety of Ying, where $q_{\mathrm{bayes}}(\ell|x,\theta)$ is $q(\ell|x,\theta)$ in Fig. 1.

It further follows from Figs. 2(c) and 2(d) that the role of Yang is not just serving the demands of Ying but also vigorously adapt its external world via visualizing the fitness of Ying to samples and also selecting the best inner encodings to enhance the performance of Ying. A modern perspective of this nature is mathematically expressed as a variety preservation principle given

by Eq. (27) in Ref. [1]. Also, maximizing $H(p||q)$ by Eq. (1) subject to Eq. (27) in Ref. [1] will give the variety preservation principle another variant as follows:

$$\begin{aligned} p\left(R|X\right) &= q\left(R|X\right)\chi_{R\in D^{\rho}_{R^*}}, \\ q\left(R|X\right) &= \frac{q\left(X|R\right)q(R)}{\int_{R\in D^{\rho}_{R^*}(X)} q\left(X|R\right)q(R)\mathrm{d}R}, \\ D^{\rho}_{R^*}\left(X\right) &= \left\{R : q\left(R|X\right)+\rho \geqslant q\left(R^*|X\right)\right\},\ \rho \geqslant 0, \\ R^* &= \arg\max_R \left[q\left(X|R\right)q(R)\right], \end{aligned} \tag{32}$$

where $D^{\rho}_{R^*}\left(X\right)$ is called apex zone or climax neighborhood, and $\chi_{u\in D}$ is a characteristic function as follows:

$$\chi_{u\in D} = \begin{cases} 1, & \text{for } u \in D, \\ 0, & \text{for } u \notin D. \end{cases} \tag{33}$$

Yang adapts samples via selecting $R \in D^{\rho}_{R^*}\left(X\right)$ to supply Ying in order to further enhance the fitness of Ying to the samples. This $D^{\rho}_{R^*}\left(X\right)$ is controlled by a scalar ρ to form a spectrum ranging between $p(R|X) = \delta(R-R^*)$ at $\rho = 0$ and $p(R|X) = q(R|X)$ for any R when $\rho > 0$ becomes large enough.

For the problem in Fig. 1, we have

$$\begin{aligned} p(\ell|x,\theta) &= q(\ell|x,\theta)\chi_{\ell\in J^{\kappa}_t}, \\ q(\ell|x,\theta) &= \frac{\alpha_\ell G(x|\mu_\ell,\Sigma_\ell)}{\sum_{j\in J^{\kappa}_t}\alpha_j G(x|\mu_j,\Sigma_j)}, \end{aligned} \tag{34}$$

where J^{κ}_t is given in Eq. (14), from which we get $p_{\ell,t}$ given by one of Eqs. (10), (14), (15), and (16).

For $Y = \{y\}$ of vectors and θ of real parameters, it is usually difficult to get $p(R|X)$ by Eq. (32). Instead, we consider each component distribution of $p(R|X)$ by its statistics up to the second order. E.g., we consider the following regression functions (the 1st order statistics):

$$\begin{aligned} &\eta_Y(X) = E_{p(Y|X,\theta)}(Y),\ \eta_\theta(X) = E_{p(\theta|X)}(\theta), \\ &\text{where} \quad E_{p(u)}(u) = \int p(u)u\mathrm{d}u, \end{aligned} \tag{35}$$

and the covariance matrices (the 2nd order statistics):

$$\begin{aligned} \Gamma^{p}_{Y|X,\theta} &= \mathrm{Var}_{p(Y|X,\theta)}(Y), \\ \Gamma^{p}_{\theta|Y} &= \mathrm{Var}_{p(\theta|Y)}(\theta), \end{aligned} \tag{36}$$

where

$$\mathrm{Var}_{p(u)}(u) = \int p(u)[u-E_{p(u)}(u)][u-E_{p(u)}(u)]^{\mathrm{T}}\mathrm{d}u.$$

That is, $p(Y|X,\theta)$ is approximated by a Gaussian $G(Y|\eta_Y(X),\Gamma^{p}_{Y|X,\theta})$ and $p(\theta|X)$ is approximated by a Gaussian $G(\theta|\eta_\theta(X),\Gamma^{p}_{\theta|Y})$.

There can be different choices for the structures of $\eta_Y(X)$ and $\eta_\theta(X)$. One is free of structure such that $\eta_Y(X)$ and $\eta_\theta(X)$ can be determined by maximizing $H(p||q)$ by Eq. (1), e.g., $\eta_Y(X) = \arg\max_Y \ln\left[q\left(X|R\right)q(R)\right]$, and $\eta_\theta(X) = \arg\max_\theta \ln\left[q\left(X|R\right)q(R)\right]$. As to be further addressed in Sect. 4.3, this choice is helpful for understanding but may lead to some implementation problem.

Instead, $\eta_Y(X)$ also takes a parametric structure that is typically decomposed into a set of simplified regressions $\{\eta_{y_t}(x_t)\}$ with

$$\begin{aligned} &\eta_Y(X,\Phi) = \{\eta_y(x_1,\Phi),\eta_y(x_2,\Phi),\ldots,\eta_y(x_t,\Phi),\ldots\}, \\ &\eta_y(x_t,\Phi) = f(\bar{y}_t) = [f(\bar{y}^{(1)}_t), f(\bar{y}^{(2)}_t),\ldots,f(\bar{y}^{(m)}_t)]^{\mathrm{T}}, \\ &\bar{y}_t = \omega_1(Wx_t+w)+\omega_2\eta^*_{y_t}(x_t)+\omega_3\eta^{\mathrm{old}}_{y_t}(x_t), \\ &\omega_j \geqslant 0,\ \sum_j \omega_j = 1,\ \Phi = \{W,w,\{\omega_j\}\}, \\ &\eta^*_y(x_t) = \arg\max_y \ln\left[q(x_t|y,\theta_{x|y})q(|\theta_y)\right], \end{aligned} \tag{37}$$

where $f(r)$ is a scalar function. Typically, we let $f(r) = r$ for an element $y^{(j)}$ that takes a real value and a sigmoid function $f(r)$ for an element $y^{(j)}$ that takes a binary value. For the factor analysis by Eqs. (25) and (26), we simply have $\omega_1 = 1, \omega_2 = 0, \omega_3 = 0$ such that

$$\bar{y}_t = Wx_t + w. \tag{38}$$

For the covariance matrices in Eq. (36), it also follows from the variety preservation principle by Eq. (32) that we consider

$$\begin{aligned} \Gamma^{p}_{Y|X,\theta} &= \Gamma^{q}_{Y|X,\theta} = \Pi^{q\ -1}_{Y|X,\theta}, \\ \Pi^{q}_{Y|X,\theta} &= -\frac{\partial^2 \ln\left[q\left(X|R\right)q\left(R\right)\right]}{\partial\mathrm{vec}[Y]\partial\mathrm{vec}[Y]^{\mathrm{T}}}, \\ \Gamma^{p}_{\theta|X} &= \Gamma^{q}_{\theta|X} = \Pi^{q\ -1}_{\theta|X}, \\ \Pi^{q}_{\theta|X} &= -\frac{\partial^2 \ln\int q\left(X|R\right)q\left(R\right)\mathrm{d}Y}{\partial\mathrm{vec}[\theta]\partial\mathrm{vec}[\theta]^{\mathrm{T}}}, \end{aligned} \tag{39}$$

which refines $\Pi^{q}_{Y|X},\Pi_\theta$ in Eq. (31) of Ref. [1] such that the conditioning part $Y|X,\theta,\theta|X$ is clarified. The integral over Y is either analytically solved or handled by the following Laplace approximation:

$$\begin{aligned} p(x) &= \int q(x|u)q(u)\mathrm{d}u \\ &\approx (2\pi)^{0.5d_u}|\Pi(u)|^{-0.5}q(x|u^*)q(u^*), \\ u^* &= \arg\max_u \ln[q(x|u)q(u)], \\ \Pi(u) &= -\frac{\partial^2\ln[q(x|u)q(u)]}{\partial u\partial u^{\mathrm{T}}}, \end{aligned} \tag{40}$$

where d_u is the dimension of u.

In addition to Eq. (23), we may have an alternative factorization of $p(R|X)$ as follows:

$$p(R|X) = p(Y|X)p(\theta|Y,X), \tag{41}$$

for which we consider

$$\begin{aligned} p\left(Y|X\right) &= q\left(Y|X\right)\chi_{Y\in D^{\rho}_{Y^*}}, \\ D^{\rho}_{Y^*} &= \left\{Y : q\left(Y|X\right)+\rho \geqslant q\left(Y^*|X\right)\right\}, \\ p(\theta|Y,X) &= q(\theta|Y,X)\chi_{\theta\in D^{\rho}_{\theta^*}}, \\ D^{\rho}_{\theta^*} &= \left\{\theta : q(\theta|Y,X)+\rho \geqslant q(\theta^*|Y,X)\right\}, \\ q(Y|X) &= \frac{\int q(X|R)q(R)\mathrm{d}\theta}{\int q(X|R)q(R)\mathrm{d}R}, \end{aligned}$$

$$q(\theta|Y,X)=\frac{\int q(X|R)q(R)\mathrm{d}Y}{\int q(X|R)q(R)\mathrm{d}\theta}, \tag{42}$$

where ρ takes a similar role to the one in Eq. (32), with different values resulting in different sizes of apex zone. Also, ρ may take different values in $D^{\rho}_{Y^*}$, $D^{\rho}_{\theta^*}$, and $D^{\rho}_{R^*}$.

Moreover, getting $q(Y|X)$ and $q(\theta|Y,X)$ involves the integral over Y and the integral over θ, which can be handled for those structures of $q(X|R)$ and $q(R)$ that are integrable over either or both Y and θ. E.g., for a Gaussian mixture in Fig. 1, with $q(X|R)=G(x|\mu_j,\Sigma_j)$ and $q(\alpha)$, $q(\mu_j)$, and $q(\Sigma_j)$ respectively in the DNW priors, these integrals can be handled analytically via its conjugate Bayes posteriori counterparts [7].

Further investigation may be made on handling the integrals that are not analytically trackable. Similar to Eq. (39), we consider

$$\begin{aligned}
&\Gamma^{p}_{Y|X}=\Gamma^{q}_{Y|X}=\Pi^{q\ -1}_{Y|X},\\
&\Pi^{q}_{Y|X}=-\frac{\partial^2\ln\int q\left(X|R\right)q\left(R\right)\mathrm{d}\theta}{\partial\mathrm{vec}[Y]\partial\mathrm{vec}[Y]^{\mathrm{T}}},\\
&\Gamma^{p}_{\theta|Y,X}=\Gamma^{q}_{\theta|Y,X}=\Pi^{q\ -1}_{\theta|Y,X},\\
&\Pi^{q}_{\theta|Y,X}=-\frac{\partial^2\ln[q\left(X|R\right)q\left(R\right)]}{\partial\mathrm{vec}[\theta]\partial\mathrm{vec}[\theta]^{\mathrm{T}}}.
\end{aligned} \tag{43}$$

In summary, a Yang machine is designed as a type of inverse of the Ying, subject to a variety preservation principle, in order to not only match the demands of Ying but also enhance the performance of Ying in adapting the external world. Specifically, the structure of $p(R|X)$ comes from the structure of each component in two types of factorization by Eqs. (23) and (41).

3.3 BYY coordination and all in one formulation

3.3.1 Coordinations across Ying-Yang, within Ying and within Yang

As discussed in Sect. 2.1 of Ref. [1], there are two types of methods to tackle an over-fitting problem of $\max_\theta \ln q(X_N|\theta)$. One is model selection that prunes away extra parameters in θ, while the other is learning regularization that imposes certain constraint in order to effectively reduce model complexity, without necessarily discarding extra parameters. However, such a regularization actually disturbs the purpose of model selection. That is, on the same $q(X_N|\theta)$ there is a trade-off between model selection and learning regularization.

In a BYY system, Ying machine models data, on which model selection is made, while Yang machine serves as one inverse of Ying machine. Instead of on Ying machine, regularization is indirectly imposed on Yang machine. Thus, regularization and model selection are coordinated within a BYY system without conflict, i.e., design of a BYY system is featured with a coordination between Ying-based model selection and Yang-based learning regularization, as outlined in the Box ④ of Fig. 3. Further details are referred to Sect. 2.1 in Ref. [1].

Model selection by Ying machine is also featured with another coordination between its two components $q(X|R)$ and $q(R)$. For an easy understanding, we take the factor analysis by Eqs. (25) and (26) as an example, where the task of model selection is determining the dimension m of y_t or equally the rank of A. The information about m is contained not only in $G(y_t|0,\Lambda)$ as a part of $q(R)$ but also in $G(x_t-Ay_t|0,\sigma^2 I)$ as a part of $q(X|R)$. Automatic model selection can be made via discarding the jth dimension of y_t by checking if either the jth element λ_j of Λ tends to zero or all the elements $a_{ij},\forall i$ in the jth column of A tend to zero. Also, we may coordinately check whether we have

$$\lambda_j\sum_i|a_{ij}|^r\to 0,\ r\geqslant 1. \tag{44}$$

That is, model selection can be made coordinately by the corresponding parts in both $q(X|R)$ and $q(R)$. In Ref. [3], the above two matrices A and $Y=\{y_t\}$ are regarded as a co-dim matrix pair that shares a same rank m. Such a matrix pair forms a building unit. We are thus motivated to design $q(X|R)$ and $q(R)$ coordinately into a hierarchy of such building units. Readers are referred to Sect. 2 of Ref. [3] for a systematic study.

Similarly, we may consider such a coordination for learning Gaussian mixture in Fig. 1 too. We can discard the j-th Gaussian by checking either $\alpha_j\to 0$ or $\mathrm{Tr}[\Sigma_j]\to 0$. It is even better to check if

$$\alpha_j\mathrm{Tr}[\Sigma_j]\to 0. \tag{45}$$

Moreover, learning regularization by Yang machine is featured with another coordination between its two parts $P(X)$ and $p(R|X)$. For $P(X)$ given by Eq. (20), a sample x_t is not directly input but smoothed by a Gaussian kernel $G(x|x_t,h^2I)$ with a bandwidth h. This is equivalent to add each sample with a Gaussian noise $G(x|x_t,h^2I)$. It is known that $\max_\theta \ln q(X_N|\theta)$ with noise added to samples is equivalent to a conventional learning regularization [44], where a difficulty is controlling an appropriate strength h. Differently, inputting $P(X)$ by Eq. (20) to a BYY system, we get not only a similar regularization role, but also an appropriate h during BYY harmony learning. This learning regularization is usually referred under the name of data smoothing. It has been empirically found that such a data smoothing regularization works. Readers are referred to a recent outline at the end of Sect. 4.2 in Ref. [1] and to historical remarks made in Sect. 23.7.4 of Ref. [42].

Learning regularization is also adjusted by $p(R|X)$ that regularizes Ying modeling indirectly via an

appropriate structure of $p(R|X)$, which will be further addressed as one demanding issue in Sect. 3.4.

3.3.2 All in one formulation: Unsupervised, supervised, and semi-supervised learning

Conventionally, a learning that bases on merely input samples is called unsupervised. Instead, a learning that bases on samples in input-output pairs is called supervised. There are many cases that merely a part of input samples is paired with their corresponding output samples, on which studies are made under the name of semi-supervised learning [45] in a sense that it consists of a supervised part and a unsupervised part. Readers are refereed to Sect. 4.4 in Ref. [1] for a recent outline of the studies on the BYY harmony learning along this direction.

The BYY system provides a unified framework to accommodate unsupervised, supervised, and semi-supervised learning all in one formulation. Four typical scenarios are summarized in Table 2, differing in special settings on one or more of the ingredients in a BYY system. Further details are explained as follows:

- Type H (hidden representation based) We consider each input-output pair $\xi_t \to \zeta_t$ as an assembled sample $x_t = [\xi_t, \zeta_t]$ of $X_N = \{x_t\}$ and maximize $H(p||q)$ to implement BYY harmony learning [25,35,46–48]. Specifically, $q(X_N|R) = q(\{\xi_t, \zeta_t\}|R)$ is decomposed into either of two factorizations that lead to typical supervised learning models. Readers are referred to Ref. [38] for a recent tutorial on these studies and also one further development called subspace-based function (SBF) that actually extends radial basis function (RBF) by locating Gaussian kernels on local subspaces, see Sect. 4.4 of Ref. [1] for a latest outline. These studies share the feature of getting a help from hidden representations, and thus shortly named Type H.
- Type R (inner regularization based) Given samples of $\{x_t, y_t\}$, we estimate $q(Y|\theta)$ by the samples of $\{y_t\}$ as a part of $q(R)$, and then incorporate the obtained $q(Y|\theta)$ into $H(p||q)$, e.g., as Eq. (69) in Ref. [3], the BYY harmony learning is implemented with the inner representation Y regularized by $q(Y|\theta)$, thus shortly named Type R. Here, one weak point is that each pairing information $\{x_t, y_t\}$ has not been taken in consideration directly.
- Type C (combining and constraining) One is simply the one by Eq. (12), originated from Eq. (7.14) in Ref. [26]. Its general form is given in Table 2. Also, Eq. (13) is rewritten in its general form under the name of Super-Bayes Combining. This name comes from a discussion made in one paragraph after Eq. (16), that is, $\gamma\delta_{\ell,j_t^*}$ makes the resulted $p(\ell|x_t, \theta)$ becomes a super-Bayes (see a further discussion around Eq. (46) and Item (A) in Sect. 3.4.2.). Actually, both the two combining types may be regarded as special cases of mixture-of-experts, i.e., $\beta(x_t|\varphi) = \gamma$ and $\beta(x_t|\varphi) = \gamma/(\gamma + q(x_t|\theta))$ with simply $q(\varphi) = \gamma$. Also, the parameter φ may be estimated by learning with help of a prior $q(\varphi)$. Finally, another option is also proposed in Table 2 for y_t of real values by constraining the regression function $\eta_Y(X, \Phi)$ of $p(Y|X, \theta)$ to satisfy the given pairing $X_N \to Y_N$.
- Type S (switching between two modes) The Yang machine switches between two modes. For the mode that has only x_t available, we let the regression $\eta_{y|x}(x_t)$ to be given by the parametric structure $\eta_y(x_t, \Phi)$. For the other mode that has a given input-output pair $x_t \to y_t$, we simply let $\eta_y(x_t) = y_t$. Actually, Type S is a special case of Type C, with its combination switching between two extreme ends $\gamma = 0$ and $\gamma = 1$, which can be observed from $E_{p(Y|X_N)}(y) = \gamma E_{p(Y|X_N,\theta)}(y) + (1 - \gamma)E_{\delta(Y-Y_N)}(y)$.

3.4 Historical remarks and demanding topics

3.4.1 Historical remarks

Conventionally, a model that describes observed samples X_N as generated from its inner coding is called a generative model or latent factor model since the inner coding Y is hidden behind the integral $q(X_N) = \int q(X_N|Y)q(Y)\mathrm{d}Y$. On the other hand, a model that maps X_N into its inner coding Y_N is called a representative or recognition model. Each of two model types has been widely studied in the literature of statistics and machine learning. Moreover, the maximum likelihood learning on $q(X_N) = \int q(X_N|Y)q(Y)\mathrm{d}Y$ involves the Bayes inverse $p(Y|X_N) = q(X_N|Y)q(Y)/q(X_N)$ for the mapping $X_N \to Y$. Naturally a question rises on why we still need to consider a BYY system as shown in Fig. 2.

Both types of models are parts of a BYY system. The BYY system considers not only two types jointly and systematically, but also includes other ingredients that have not been or seldom involved in the studies of either generative models or representative models.

Started from Ref. [2] in 1995, efforts in the early period were mainly made on showing that a BYY system provides a unified framework that leads to many existing learning approaches, with architectures of BYY system classified into three types. Referring to Fig. A2 in Ref. [1], main streams of efforts are outlined as follows:

- Backward architecture (B-architecture): $q(X|Y)$ and $q(Y)$ are given by parametric structures; while $p(Y|X)$ is free of structure, such that not only minimizing the Kullback-Leiber (KL) divergence

Table 2 Supervised and semi-supervised Bayesian Ying-Yang harmony learning

learning from the input-output pairs $X_N = \{x_t\}_{t=1}^N \to Y_N = \{y_t\}_{t=1}^N$

Type S **semi-supervised learning**

For a vector valued y, let $\eta_y(x_t)$ denote either $E_{p(y|x_t,\theta)}(y)$ or $E_{p(y|x_t)}(y)$.

Usually, $\eta_y(x_t) = \begin{cases} y_t, & \text{for each supervised pair } x_t, y_t, \\ \eta_y(x_t,\Phi) \text{ given by Eq. (37)}, & \text{with only } x_t \text{ available,} \end{cases}$

and
$$\mathrm{Var}_{p(y|x_t,\theta)}(y) = \Gamma^q_{y|x,\theta} = \Pi^{q\,-1}_{y|x,\theta},\ \Pi^q_{y|x,\theta} = -\frac{\partial^2 \ln[q(x\,|\,y,\theta_{x|y})q(y\,|\,\theta_y)]}{\partial y \partial y^{\mathrm{T}}},$$
$$\mathrm{Var}_{p(y|x_t)}(y) = \Gamma^q_{y|x} = \Pi^{q\,-1}_{y|x},\ \Pi^q_{y|x} = -\frac{\partial^2 \ln \int q(\theta)q(x\,|\,y,\theta_{x|y})q(y\,|\,\theta_y)\mathrm{d}\theta}{\partial y \partial y^{\mathrm{T}}}.$$

For a discrete $y = \ell$, we use $p(\ell|\,x_t,\theta)$ of Type C with $\gamma = \begin{cases} 1, & \text{for a pair } x_t, y_t, \\ 0, & \text{only } x_t \text{ available.} \end{cases}$

Type C **semi-supervised learning**

Combining:

$p(Y|\,X_N) = (1-\gamma)p(Y|\,X,\theta) + \gamma\delta(Y - Y_N)$, in general,
$p(y_t|\,x_t,\theta) = (1-\gamma)p(y|\,x_t,\theta) + \gamma\delta(y - y_t)$, per sample pair.

Super-Bayes combining:

$$p(\ell|\,x_t,\theta) = \frac{\gamma\delta_{\ell,j_t} + q(x_t|\ell,\theta_{x|y})q(y=\ell)}{\gamma + q(x_t\,|\,\theta)},\ q(x_t\,|\,\theta) = \sum_\ell q(x_t|\ell,\theta_{x|y})q(y=\ell).$$

Mixture of experts:

$p(y_t|\,x_t,\theta) = [1-\beta(x_t|\,\varphi)]p(y|\,x_t,\theta) + \beta(x_t|\,\varphi)\delta(y - y_t),\ 0 \leqslant \beta(x_t|\,\varphi) \leqslant 1.$

Constraining:

$p(Y|\,X) = G(Y\,|\,\eta_Y(X,\Phi),\Pi^{q\,-1}_{Y|X})$ by $\Pi^q_{Y|X}$ in Eq. (43).

$p(Y|\,X,\theta) = G(Y\,|\,\eta_Y(X,\Phi),\Pi^{q\,-1}_{Y|X,\theta})$ by $\Pi^q_{Y|X,\theta}$ in Eq. (39).

Subject to that the pairing $\{X_N, Y_N\}$ satisfies $\eta_Y(X,\Phi)$ in Eq. (37).

Type R **semi-supervised learning**

$q(Y\,|\,\Theta)$ is estimated from Y_N for a regularization role.

Type H **supervised learning from the input-output pairs** $\xi = \{\xi_t\}_{t=1}^N \to \zeta = \{\zeta_t\}_{t=1}^N$

$q(X\,|\,Y,\theta) = q(\xi,\zeta\,|\,Y,\theta) = q(\xi\,|\,Y,\theta_\xi)q(\zeta\,|\,Y,\theta_\zeta)$,

via a layer of hidden representation Y, which leads to a three-layer networks.

$q(X\,|\,Y,\Theta) = q(\xi,\zeta\,|\,Y,\Theta) = q(\xi\,|\,\Theta_\xi)q(\zeta\,|\,\xi,Y,\Theta_\zeta)$,

which leads to radial basis function and mixture of experts.

makes $p(Y|X_N) = q(X_N|Y)q(Y)/q(X_N)$ and leads to those existing learning approaches as indicated by the path of Box ① → Box ② → Box ③ in Fig. A2, but also maximizing $H(p||q)$ further leads to those maximum a posteriori studies, as indicated by the path of Box ① → Box ④ on Fig. A2.

- Forward architecture (F-architecture): $p(Y|X)$ and $q(Y)$ are given by parametric structures, while $q(X|Y)$ is free of structure, such that not only minimizing the KL divergence leads to those existing approaches as indicated by the path of Box ⑤ → Box ⑥ on Fig. A2, but also maximizing $H(p||q)$ further degenerates into two separated paths. One is the path of Box ⑦ → Box ② → Box ③, while the other is the path of Box ⑦ → Box ⑪.
- Bi-directional architecture (BI-architecture): both $p(Y|X)$ and $q(X|Y)$ are given by parametric structures, for which minimizing the KL divergence leads to the Helmholtz free energy or variational function as indicated by the path of Box ⑧ → Box ⑨ on Fig. A2; while maximizing $H(p||q)$ leads to a framework with a new mechanism for model selection, as indicated by the Box ⑩ on Fig. A2.

Readers are further referred to Sects. 22.9 and 23.7 of Ref. [42] for detailed historical remarks on studies made before 2003, and also to Appendix A of Ref. [1] as well as Sects. 4.2.2 and 4.2.3 of this paper for recent overviews.

The second period of studies on the BYY system is mainly featured by examining the role of each of four components $p(X)$, $p(Y|X)$, $q(Y)$, and $q(X|Y)$, the

structures for each component, and the order of considering the four components. One key point is identifying what types of dependence among samples to be modeled. As summarized Fig. 22.1 in Ref. [42], the types of dependence include modular dependence, temporal dependence, and topological dependence. Then, $q(Y)$ is designed to describe one of these types, and accordingly the structure of $q(X|Y)$ is designed. Further details are referred to Sects. 22.4 and 22.9 in Ref. [42], Sect. II in Ref. [5], and the previous two subsections in this paper.

In Ref. [6], the issue of BYY system design is further re-elaborated from a perspective of two intelligent abilities and three inverse problems. Particularly, as shown in its Tables 2–5, each of the components $p(R|X)$, $q(X|R)$, $q(R)$ are factorized into several parts with each in one of typical choices. Subsequently in Ref. [38], three design principles for a BYY system are proposed, namely *least redundancy principle* for $q(R)$, *divide and conquer principle* for $q(X|R)$, and *uncertainty conversation principle* or *variety preservation principle* for $p(R|X)$. Readers are further referred to Sects. 4.2 and 4.4 in Ref. [1] and Sect. 3.2 of this paper for latest overviews and particularly for further developments on semi-supervised learning, and also to Sects. 2 and 4 of Ref. [3] for a new configuration of the BYY system featured with a hierarchy of co-dim matrix pairs.

3.4.2 Demanding topics

Next, we summarize challenging issues that wait to be solved and future topics that deserve to be explored.

• On Ying structure design

(a) As addressed at the beginning of Sect. 3.2.1, the first topic is featured by different choices that trade off the roles between $q(Y)$ and $q(X|R)$ under the guideline of a least complexity principle. The structure of $q(Y)$ and the structure of $q(X|R)$ are considered according to what types of dependence there are within the $d \times N$ data matrix $X_N = [x_1, x_2, \ldots, x_N]$ and its corresponding inner representation $Y_N = [y_1, y_2, \ldots, y_N]$ of an $m \times N$ data matrix. We consider a post bi-linear system $X = \eta^x + E$ by Eq. (10) in Ref. [3] with η^x coming from an element-wise monotonic scalar mapping from $\tilde{X}$ Typically, $\tilde{X}$ comes from linear map of Y, with two typical examples as follows:
   - $\tilde{x}_t = Ay_t$ is mapped homogenously per column and y_t is inter-dimensional independent, as given by Eq. (22) in Ref. [3], i.e., Y_N is independent across its rows, while the dependence cross the rows of X_N is modeled by A;
   - The dependence cross the columns of X_N is modeled by certain dependence structure across the columns of Y_N, e.g., a line structure (i.e., temporal dependence) by Eq. (30) or a graph Laplacian by Eq. (107).

   Usually, the row dependence of X_N and column dependence of X_N are assumed to be decomposable, with the former modeled by $q(X|R)$ and the latter by $q(Y)$. Following the least complexity principle, an informal discussion has been made after Eq. (30) to support this treatment in a special case. Generally, it remains to be a challenge issue to investigate under which situations this decomposition assumption can be justified and whether such a formulation on the structures of $q(X|R)$ and $q(Y)$ entertains the least complexity principle.

(b) The FA by Eqs. (25) and (26) is one special case of the above formulation. As discussed after Eq. (27), the linearity of Ay_t leads to an equivalent family of $q(Y)$ structures in term of the maximum likelihood learning, i.e., the FA-D family by Eq. (27) in Ref. [3]. In term of model selection (e.g., determining the row dimension of X_N), however, there is at least an optimal one that is better than others. This nature has been verified by extensive experiments on FA-a versus FA-b in Ref. [34] and also analytically justified by discussions made after Eqs. (28) and (29) in Ref. [3]. Generally, such a nature should be applicable to the above general post bi-linear system too. Further investigations are deserved on comparing the model selection performances of the following paired models:
   - Local factor analysis (LFA) by Eqs. (28) and (29) versus a mixture of factor analysis featured with $y_{t,\ell} \sim G(y|0, I)$ as used in Refs. [27,36,37];
   - Temporal FA (TFA) by Eqs. (25) and (30), versus its counterpart featured with $\varepsilon_t \sim G(\varepsilon_t|0, I)$, which will be further discussed in Sect. 5.2.1 of this paper.
   - Manifold learning with $q(Y)$ by Eq. (66) in Ref. [3] versus its counterpart featured with $q(Y)$ by Eq. (107), which will be further discussed in Sect. 5.2.1 of this paper.

(c) Efforts also deserve to be made beyond the above formulation. Beyond a row-column decomposition, we may consider the dependence among elements of X_N in other choices. Instead of $\tilde{x}_t = Ay_t$, we may generally consider $\text{vec}[\tilde{X}] = \mathcal{A}\text{vec}[\tilde{Y}]$, which degenerates to $\tilde{x}_t = Ay_t$ for a special structure $\mathcal{A} = \text{diag}[A, \ldots, A]$. There need further efforts on explore other special structures of $\mathcal{A}$. Also, we may proceed beyond that $\tilde{X}$ comes from linear map of Y. E.g., $\tilde{x}_t$ is a quadratic regression from y_t by Eq. (80) in Sect. 5.1 of this paper, which is helpful to verify whether the interaction between the corresponding two SNPs in GWA studies (see Sect. 6.1 of this paper). Following the least complexity

principle, we prefer that each quadratic term is taken in consideration only when we have to. Hence, efforts are needed to seek a learning mechanism that extra parameters are pushed towards zeros such that the coefficients of quadratic terms are pushed more strongly than the coefficients of linear terms.

(d) Further investigation is also needed on the structure of inner representation $q(R) = q(Y)q(\theta)$ by Eq. (21). In general, each sample x_t is encoded by both a discrete representation and a real inner representation. One example is LFA by Eqs. (28), (29), and (21), where we have $q(Y_N) = q(\{\ell_t, y_t\}_{t=1}^N)$. We may further extend LFA to model temporal/column dependence among X_N as follows:
   - Modeling temporal dependence among $\{y_t\}_{t=1}^N$ by Eq. (30) but still regarding $\{\ell_t\}_{t=1}^N$ as i.i.d. samples, each FA is replaced by TFA, by which $\{x_t\}_{t=1}^N$ is regarded as a mixture of several stationary process;
   - Modeling temporal dependence among $\{\ell_t\}_{t=1}^N$ by a Markov chain but still regarding $\{y_t\}_{t=1}^N$ as i.i.d. samples, we get a hidden Markov model with each of hidden states associated with a FA model, by which $\{x_t\}_{t=1}^N$ is regarded as a temporal process that switches across several clusters of i.i.d. samples.
   - Modeling temporal dependence among $\{y_t\}_{t=1}^N$ by Eq. (30) and among $\{\ell_t\}_{t=1}^N$ by a Markov chain, we are lead to an HMM gated TFA modeling that will be further introduced in Sect. 5.2.2 and Fig. 9, by which $\{x_t\}_{t=1}^N$ is regarded as a nonstationary process that switches across different stationary segments. Moreover, the hidden states are connected in one of the structures shown in Fig. 10, to ensure the unidirectional nature and to probabilistically describe a random length per segment.

   We need a comparative study on the above ways of temporal modeling, and also a further investigation on the coordination role of an appropriate prior $q(\theta)$ in $q(R)$, e.g., a comparative study on Jeffreys priors versus the so-called induced bias cancellation (IBC), see Eq. (21) in Ref. [39], Sect. II(A) of Ref. [40], and the last part of Sect. 4.2 in Ref. [1].

• On Yang structure design

(A) As addressed at the beginning of Sect. 3.2.2, the task of Yang structure design is mainly designing the probabilistic structure of $p(R|X)$, under the guideline of the variety preservation principle. We need a measure $U(p)$ to describe the variety of a distribution p, e.g., using either Shannon or Renyi entropy as shown in Fig. 4(c) in Ref. [38]. In Fig. 1, we say $p(\ell|x,\theta)$ is more selective than $\tilde{p}(\ell|x,\theta)$ if $U(p(\ell|x,\theta)) < U(\tilde{p}(\ell|x,\theta))$, or $\tilde{p}(\ell|x,\theta)$ is less selective than $p(\ell|x,\theta)$. Moreover, we take the Bayes inverse of Ying machine as a standard reference. We say a structure of $p(R|X)$ is super-Bayes if it is more selective than this standard reference, or inversely sub-Bayes if it is less selective than the reference. One extreme end $E_{\rm sup}$ of super-Bayes structures is $p(R|X) = \delta(R - R^*)$, while one other extreme end $E_{\rm sub}$ of sub-Bayes structures is a uniform distribution of $p(R|X)$ on its supporting domain. As a whole, the possible structures of $p(R|X)$ could be a spectrum

$$E_{\rm sup} \longleftrightarrow Bayes \cdot inverse \longleftrightarrow E_{\rm sub}. \qquad (46)$$

Towards $E_{\rm sup}$, it becomes easier to handle the integral over R (e.g., θ, Y) and the integral actually disappears at $E_{\rm sup}$ with $p(R|X) = \delta(R - R^*)$. Also, it tends to consider a few representations around R^* such that Ying has a best fitting on the given sample set X_N while it is weak on generalization (i.e., fitting new data from a same underlying regularity). Away from $E_{\rm sup}$ towards $Bayes \cdot inverse$ and then further to $E_{\rm sub}$, computing cost increases and fitting error trades off improvements on generalization, for which further investigations are needed. Efforts are also needed on seeking a good measure for variety and on whether $Bayes \cdot inverse$ provides an optimal structure of $p(R|X)$

(B) Existing studies have been mainly made on $p(R|X)$ within the range $E_{\rm sup} \longleftrightarrow Bayes \cdot inverse$, supported by an empirical finding that an optimal structure likely falls within this range. It is not clear whether the range $Bayes \cdot inverse \longleftrightarrow E_{\rm sub}$ is useful. Some experiments show that it improves generalization to initialize $p(R|X)$ within the range $E_{\rm sub} \longleftrightarrow Bayes \cdot inverse$ and gradually move into the range $E_{\rm sup} \longleftrightarrow Bayes \cdot inverse$ towards $E_{\rm sup}$. Further investigations are needed on these empirical findings. Controlling the structure of $p(R|X)$ in such a way is also named structural regularization since it regularizes the Ying machine indirectly via the structure of $p(R|X)$. Further efforts are needed on how this structural regularization is appropriately controlled in a coordination with Ying-based model selection and also in a coordination with data smoothing regularization via Eq. (20) by controlling an appropriate strength h.

(C) The structure of Bayes inverse is featured by $p(R|X) \propto q(X|R)q(R)$, that is, the Yang has a linear relation with the Ying. One way of controlling $p(R|X)$ to vary among the spectrum $E_{\rm sup} \longleftrightarrow Bayes \cdot inverse \longleftrightarrow E_{\rm sub}$ is extending this linear relation to become nonlinear. For an example, in Fig. 1 we may let $p(\ell|x,\theta) \propto q_{\rm bayes}^{1/\lambda}(\ell|x,\theta)$, see Eq. (23.50) in Ref. [42], which varies within

$E_{\sup} \longleftrightarrow Bayes \cdot inverse$ if $\lambda \geqslant 1$, and varies within $Bayes \cdot inverse \longleftrightarrow E_{\sub}$ if $1 > \lambda > 0$. In general, we consider $p(R|X) \propto [q(X|R)q(R)]^{1/\lambda}$. Alternatively, we may also keep the linear relation $p(R|X) \propto q(X|R)q(R)$ merely within an apex zone, e.g., one of Eqs. (32), (34), and (42), incurring a nonlinearity when $p(R|X)$ is cut-off to zero outside of the apex zone. Accordingly, we control the size of the apex zone for controlling $p(R|X)$ to vary among $E_{\sup} \longleftrightarrow Bayes \cdot inverse$. Further investigation deserves to be made on both of these issues.

(D) Both the factorization by Eq. (23) and the factorization by Eq. (41) have only been partially studied yet. Most of existing studies on Eq. (23) use $p(\theta|X)$ approximately with $\Pi^q_{\theta|Y,X}$ as $\Pi^q_{\theta|X}$ in the place of the one in Eq. (39). To be addressed in Sect. 4.2 of this paper, we may get an improvement by directly handing the integral in Eq. (39) either analytically or by means of Eq. (40). On the other hand, efforts have been made on solving the integrals in Eq. (42) analytically for a Gaussian mixture, with $q(X|R) = G(x|\mu_j, \Sigma_j)$ and $q(\alpha)$, $q(\mu_j)$, and $q(\Sigma_j)$ respectively in DNW priors [7]. Further investigation may be made on handling the integrals that are not analytically trackable. E.g., similar to Eq. (39), we consider Eq. (43).

(E) Finally, semi-supervised learning comes from a special structure of $p(Y|X_N, \theta)$ that simply combines a supervised teaching pairing $\delta(Y - Y_N)$ with a Bayes inverse type structure. Several choices are summarized in Table 2 under Type C. A comparative study needs to be further conducted on these types of combinations. Further investigation is also needed on an appropriate value of γ, which controls the supervised strength and may also be learned via maximizing $H(p||q)$ with help of an appropriate prior $q(\gamma)$, e.g., a beta distribution.

4 Topics on BYY harmony learning

4.1 Measuring bi-entity proximity

4.1.1 Harmony functional: A unified scheme

As outlined in Fig. 4, we start from typical measures for bi-entity proximity (see Box ①). The task of learning is making a learner (a parametric model) to describe the regularity underlying a set of samples in a world of the learner's observation, under the guidance of a learning principle or theory that measures a proximity of the regularity described by the model to the regularity underlying samples, or how the learner's behavior is close to what observed in its world.

Mathematical formulation of a proximity measure depends on which space is considered. In an Euclidean space and a Hilbert space, a proximity between two entities could be measured from either a perspective of best agreement (e.g., equal, inner-product, similarity, correlation, projection), or a perspective of least difference (e.g., error, residuals, distance). Two perspectives are closely related but usually different though they become equivalent in certain special cases. Further details are referred to Appendix A of Ref. [1].

For learning by probabilistic models, we consider entities in a probability space or generally a measure space. Let P, Q, μ to be σ-finite measures on the same measure space (X, Σ), we observe the proximity between P, Q under μ as a common background. With help of Radon-Nikodym derivatives [49], we consider a product $f(\mathrm{d}Q/\mathrm{d}\mu)\mathrm{d}P/\mathrm{d}\mu$ that measures every local agreement between P, Q calibrated by μ, with a scalar function $f(r)$ that indicates Q in a primary consideration. This $f(r)$ has the following two natures:

- $f(r)$ monotonically increases with r and $\mathrm{d}^2 f(r)/\mathrm{d}r^2 < 0$ such that the contributions from those local regions with large values of $\mathrm{d}Q/\mathrm{d}\mu$ are appropriately discounted, and that there could be a large dynamic range on which the behavior of Q can be evaluated more homogenously.
- Taking those local regions in consideration everywhere, we have the harmony functional $H_\mu(P||Q)$ given in Fig. 5, which is triple-relation among $\mathrm{d}P$, $\mathrm{d}Q$, and $\mathrm{d}\mu$. The scalar function $f(r)$ should have a matching nature such that $\max_q H_\mu(P||Q)$ subject to a given p pushes q towards to p or preferably reaching $p = q$. One typical example is

$$f(r) = \ln r. \tag{47}$$

In such a setting, Q takes a primary role with a large behaving range that can be evaluated in details, while P takes a secondary role that maximizing $H_\mu(P||Q)$ makes P to concur with or harmonize to Q on those major regions that Q behaves.

Oppositely, if we are given one $f(r)$ that monotonically increases with r and $\mathrm{d}^2 f(r)/\mathrm{d}r^2 > 0$, such that focuses are put on those local regions with large values of $\mathrm{d}Q/\mathrm{d}\mu$. Reversely, P becomes a primary consideration while Q takes a secondary role that coheres to P. This abnormal case is less useful since it concentrates only on those regions with large values of $\mathrm{d}P/\mathrm{d}\mu$.

Generally, $H_\mu(P||Q)$ is a functional of P, Q, μ as a measure for a proximity between two entities P, Q from a best agreement perspective with a common background μ. When $f(r) = r$, $H_\mu(P||Q)$ is also an inner product in the Hilbert space, since P, Q belonging to the probability space implies that P, Q belong to the class L_2. Generally, when $f(r) \neq r$, $H(p\,\|q)$ may be neither an inner product of p and q since q and $f(q)$ are not exchangeable

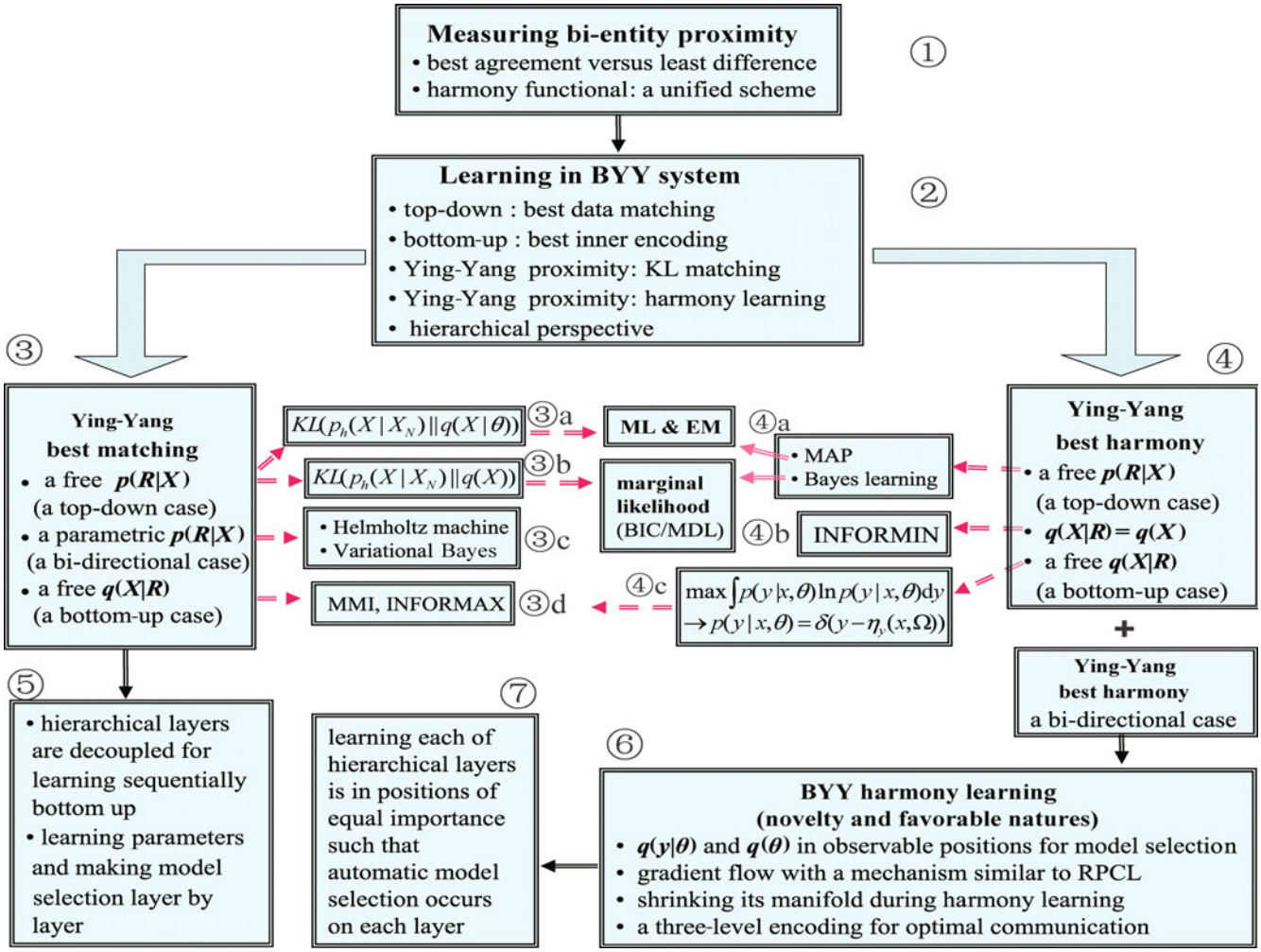

Fig. 4 Theory of Best Ying-Yang harmony

nor an inner product of p and $f(q)$ because $f(q)$ may not belong to the class L_2 especially when q is on an infinite support. As suggested in Sect. III of Ref. [4], such an asymmetric inner-product can be regarded as a special case of a general concept "projection" that is no longer symmetric, e.g., projecting one vector onto another vector in the Euclidean space. $H_\mu(P||Q)$ may be regarded as a generalized projection in a measure space.

Interestingly, as shown in Fig. 5, $H_\mu(P||Q)$ provides a unified scheme that covers the following three branches:

- When $Q = P$ in a complete match, $H_\mu(P||Q)$ becomes a bi-relation $H_\mu(P||P)$ as shown in Box ③a, Box ③b, and Box ③c, which describes the compactness of the configuration of $\mathrm{d}P/\mathrm{d}\mu$ or density $p(x)$. For finite measures P, μ, maximizing $H_\mu(P||P)$ makes $\mathrm{d}P/\mathrm{d}\mu$ or $p(X)$ more compacted or concentrated.
- When $\mu = P$, $H_\mu(P||Q)$ becomes another bi-relation $H_P(P||Q)$ that describes an agreement by $f(\mathrm{d}Q/\mathrm{d}P)$ measured with $\mathrm{d}P$ (a re-scaled projection of differential ratio onto $\mathrm{d}P$) as shown in Box ②a or the negation of the well-known KL divergence in the Box ②b when $f(r) = \ln r$ and μ is Lebesgue. Here, the calibration role of μ is shut off by P, seeking a best agreement between P and Q is equivalent to seeking a least disagreement or difference between P and Q, both of which approaches $P = Q$.
- Generally, as shown in Box ①a, Box ①b, and Box ①c, maximizing a triple-relation $H_\mu(P||Q)$ consists of both maximizing $H_\mu(P||P)$ to make $\mathrm{d}P/\mathrm{d}\mu$ or $p(X)$ more compact and minimizing $KL(P||Q)$ to approach $P = Q$.

4.1.2 Why taking the name harmony functional

According to a Chinese philosophical concept called harmoniousness, a good pattern for a relation between two coexisting individuals or entities is featured with acknowledgment of individual differences (usually one primary and one secondary, being mutually complementary), respect of each own value, avoidance of mutual confrontation, and harmony with their common background world. The harmony functional $H_\mu(P||Q)$ echoes this sprit in that not only each entity has its own measure with a re-scaling $f(r)$ to signify one in a primary

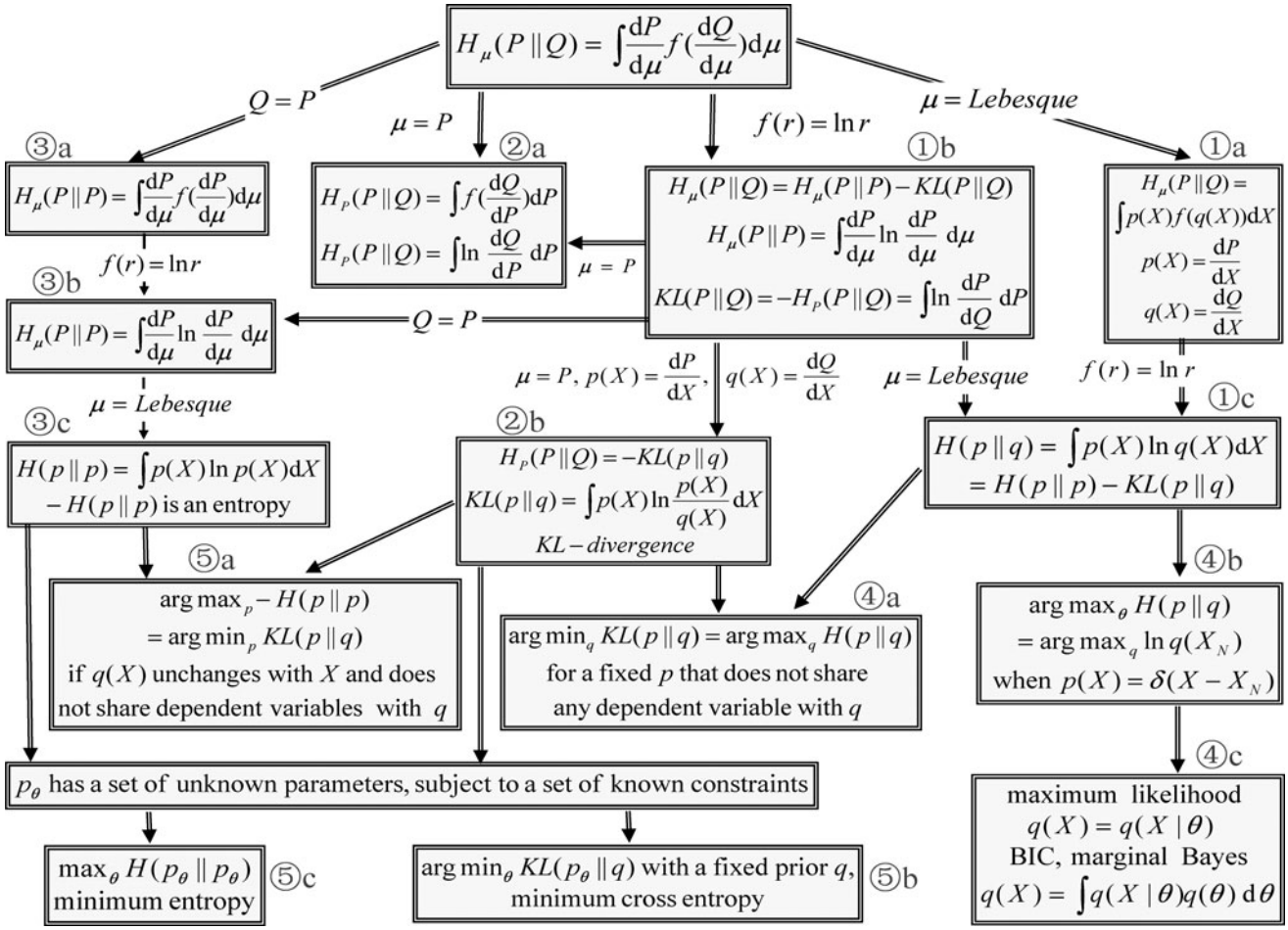

Fig. 5 Harmony functional: A unified scheme for bi-entity proximity

consideration but also two measures P, Q coordinate in a positive side because of maximizing $H_\mu(P||Q)$. Moreover, a harmonious interaction with their background world is considered via a common calibration by μ.

Also, it follows from the discussion on the 2nd column of page 299 in Ref. [1] that one must not confuse $H(p||q)$ with a terminology called cross entropy, used in the literature of signal processing and information theory under the name of minimum cross entropy (MCE). Actually, the name "cross entropy" mixed up two scenarios:

- One is a special case of the Box ④a in Fig. 5 with a fixed reference distribution p, where $\max H(p||q)$ and $\min KL(p||q)$ with respect to q equivalently leads to $q = p$. Moreover, if the fixed p is simply empirical distribution, it becomes equivalent to maximum likelihood (ML), marginal Bayes, and BIC, as shown by the Box ④b and the Box ④c in Fig. 5.
- The other scenario is to optimize p against a fixed q, for which $\max_p H(p||q)$ leads to $p(x) = \delta(x-c)$ if p is free of constraint. This was regarded as a useless degenerated case, and thus no further effort has been made along this direction. Instead, the MCE studies have been widely made on $\min_p KL(p||q)$ with a fixed q subject to a set of known constraints on p [8], as illustrated by the Box ⑤a and the Box ⑤b in Fig. 5. Moreover, as illustrated by the Box ⑤b, it includes the maximum entropy approach as a special case [9,10]. Also, the Box ③c leads to the minimum entropy approach as a special case in the Box ⑤c.

With a fixed q, $\max_p H(p||q)$ and $\min_p KL(p||q)$ are not equivalent, and thus referring both of them by the MCE name had ever created some confusion. In the literature of signal processing and information theory, there have been already some authors attempting to resolve the inconsistency by reassigning the terminology "cross entropy" to merely indicating $KL(p||q)$.

Therefore, we name $H(p||q)$ as a harmony functional instead of cross entropy. Not only $H_\mu(P||Q)$ provided a unified paradigm that covers other special cases, but also the above mentioned useless nature $p(x) = \delta(x-c)$ of $\max_p H(p||q)$ becomes useful and important when p, q are given by a Ying-Yang system, which will be further discussed in Sect. 4.2.3 (see the fourth aspect).

4.2 Best Ying-Yang harmony principle

4.2.1 Measuring unidirectional proximity

Considering the Bayesian Ying-Yang system in Fig. 2(a),

measuring bi-entity proximity can be handled from different perspectives, as outlined by the Box ② in Fig. 4. Existing efforts usually consider bi-entity proximity from a unidirectional perspective, that is, either a top-down direction or a bottom-up direction.

From a top-down direction, the proximity between a system and a set X_N of samples is considered via $q(X) = \int q(X|R)q(R)\mathrm{d}R$ for a best match to $p(X) = \delta(X - X_N)$ by $H(p||q) = \int p(X)\ln q(X)\mathrm{d}X = \ln q(X_N)$. Two typical examples are outlined by the Box ④b and the Box ④c in Fig. 5. One is $q(X|\theta) = \int q(X|Y, \theta_{x|y})q(Y|\theta_y)\mathrm{d}Y$, widely studied under the name of the maximum likelihood learning for a latent modeling, while the other is $q(X|k) = \int q(X|\theta)q(\theta)\mathrm{d}\theta$ that was previously used for developing the BIC criterion [11] and intensively studied under the name of the marginal likelihood based Bayes studies in the literature of machine learning [12–14].

From a bottom-up direction, the proximity is considered at getting $p(Y) = \int p(Y|X)p(X)\mathrm{d}X$ for a best inner encoding that matches a structural specification of $q(Y)$, typically by minimizing $KL(p||q)$. As outlined in the Box ⑤b of Fig. 5, one instance is the previous discussed MCE with q being a fixed prior. The other instance is that $q(Y)$ is mutually element-wise independent, which leads to the studies under the name of the minimum mutual information (MMI) [15]. Specifically, one limit case is the maximum information (INFORMAX) [16] when $q(Y)$ is uniform or noninformative. Both MMI and INFORMAX have been widely adopted in the studies of independent component analysis (ICA) [17]. With $p(Y)$ and $p(Y|X)$ subject to some constraints, we may also estimate unknown parameters of p via minimizing entropy or INFORMIN, as listed in the Box ⑤c of Fig. 5.

In addition to observing the proximity at the ingredient $q(Y)$, studies from a bottom-up direction also include observing the proximity to the ingredient $p(Y|X)$. We may use $\delta(Y - Y_N(X_N))$ to represent a pairing Y_N, X_N of samples. The proximity between the ingredient $p(Y|X)$ and the pairing of samples is considered via maximizing $H(\delta(Y - Y_N(X_N))||p(Y|X)) = \ln p(Y_N|X_N)$. Particularly, we consider $p(Y|X) \propto q(X|Y, \theta_{x|y})q(Y|\theta_y)$ when $q(X|Y, \theta_{x|y})$ and $q(Y|\theta_y)$ are available. In the literature of speech recognition, maximizing $\ln p(Y_N|X_N)$ is studied also under the name of the maximum mutual information (MMI) as a discriminative training criterion [18,19], which is actually different from the above mentioned MMI [15] though a same name is used.

Given $X_N = \{X^{(1)}, X^{(2)}, \ldots, X^{(N)}\}$ as an observation sequence and $Y_N = \{Y^{(1)}, Y^{(2)}, \ldots, Y^{(N)}\}$ as the corresponding word-sequence / phone-sequence / state-sequence, i.e., the segment $X^{(r)}$ corresponds to the word / phone / state $Y^{(r)}$, the detail formulae of this discriminative training MMI criterion and its further extensions [50–53] are given as follows:

$$
\begin{aligned}
F_{\mathrm{MMI}}(Y_N, X_N, \theta) &= \sum_{r=1}^{R} \ln p(Y^{(r)}|X^{(r)}, \theta), \\
F_{\mathrm{MCE}}(Y_N, X_N, \theta) &= \sum_{r=1}^{R} s\left(1 - \frac{1}{p(Y^{(r)}|X^{(r)}, \theta)}\right), \\
F_{\mathrm{FPE}}(Y_N, X_N, \theta) &= \sum_{r=1}^{R}\sum_{Y} p(Y|X^{(r)}, \theta)L(Y, Y^{(r)}), \\
p(Y^{(r)}|X^{(r)}, \theta) &= \frac{q(X^{(r)}|Y^{(r)}, \theta_{x|y})q(Y^{(r)}|\theta_y)}{\sum_{Y\in D_Y^{(r)}} q(X^{(r)}|Y, \theta_{x|y})q(Y|\theta_y)},
\end{aligned}
\tag{48}
$$

where $D_Y^{(r)}$ is a set of candidates that $X^{(r)}$ may be classified into, and $s(r)$ is a sigmoid function, and $L(Y, Y^{(r)})$ is the loss function of word sequence Y against the reference $Y^{(r)}$.

Each of the above studies is featured by measuring a bi-entity proximity along one direction. Applying to the Bayesian Ying-Yang system in Fig. 2(a), such a bi-entity proximity only focuses on a part of the system while lacks of an appropriate coordination with other parts within the system. One way to improve is combining a top-down measure and a bottom-up measure to jointly consider bi-entity proximity at more than one parts. However, an inappropriate combination may produce some inconsistency.

4.2.2 Ying-Yang best matching: A unified perspective

A system oriented learning principle is obtained by measuring bi-entity proximity between Ying machine $q(X|R)q(R)$ and Yang machine $p(R|X)p(X)$, for which we consider the harmony functional $H_\mu(P||Q)$ in Fig. 5, where the first branch considers a degenerated case $Q = P$ that needs not to be discussed separately, but will be covered during discussions on other branches.

As outlined by the Box ②a and the Box ②b in Fig. 5, the second branch is featured with $\mu = P$ such that maximizing $H_P(P||Q)$ or equivalently minimizing the KL divergence $KL(P||Q)$ targets at $P = Q$, which is shortly named as Ying-Yang best matching or Bayesian Ying-Yang (BYY) best matching.

To get some insights, we start from the standard KL divergence given by the Box ②b in Fig. 5, namely we move to the Box ③ in Fig. 4. According to Appendix A of Ref. [1] and particularly the road map given by its Fig. A2, this BYY best matching acts as a general framework that unifies existing learning principles as follows:

- Minimizing $KL(p(Y|X)p(X)||q(X|Y)q(Y))$ with respect to a structure free $p(Y|X)$ leads to $KL(p(X)||q(X|\theta))$, as shown by the Box ③a in Fig. 4. When $p(X) = \delta(X - X_N)$, we are lead to the maximum likelihood on a latent model by $q(X|\theta) = \int q(X|Y)q(Y)\mathrm{d}Y$, for which the implementation by Eq. (18) degenerates into the well

known EM algorithm [2]. Moreover, we are lead to their extensions with a data smoothing regularization when $p(X) = p_h(X|X_N)$ by Eq. (20).

- Minimizing $KL(p(\theta|X)p(X)||q(X|\theta)q(\theta))$ with respect to a free $p(\theta|X)$ leads to minimizing $KL(p(X)||q(X))$, as shown by the Box ③b in Fig. 4. When $p(X) = \delta(X - X_N)$, we get the marginal likelihood $q(X) = \int q(X|\theta)q(\theta)\mathrm{d}\theta$, from which we are lead to the Box ④c in Fig. 5, that is, BIC [11] or MDL [54]. Again, we are lead to their extensions with a data smoothing regularization when $p(X) = p_h(X|X_N)$ by Eq. (20).
- For the Box ③c in Fig. 4, the minimization of $KL(p(Y|X)p(X)||q(X|Y)q(Y))$ with both a parametric $p(Y|X)$ and a parametric $q(X|Y)$ leads to the classic Helmholtz free energy [55] from a different perspective, while minimizing $KL(p(\theta|X)p(X)||q(X|\theta)q(\theta))$ with respect to a parametric $p(\theta|X)$ leads to a formulation that becomes equivalent to the variational Bayes [13,31].
- Minimizing $KL(p(Y|X)p(X)||q(X|Y)q(Y))$ with respect to a free structure $q(X|Y)$ leads to $\min KL\left(p(Y)\,\|q(Y)\right)$, as shown by the Box ③d in Fig. 4. Similar to the previous discussions on the Box ⑤b of Fig. 5, we are again lead to MCE, MMI, and INFORMAX.

Finally, we move to the Box ⑤ in Fig. 4. As introduced in Sect. 3.1, the inner representation $R = \{Y, \theta, k, \Xi\}$ in the Ying domain actually has a three-layer hierarchy $\{\{\{Y\}, \theta\}, k, \Xi\}$ with Y on the deepest layer and k, Ξ on the top layer. Accordingly, $q(R)$ in Eq. (21) and $p(R|X)$ in Eq. (23) may also be considered in such a hierarchy. As introduced in Sect. 5.2 of Ref. [1], Ying-Yang best matching has a bottom-up decoupling nature that makes the tasks of learning hierarchical layers be decoupled sequentially bottom-up such that the tasks of handling latent variables, parameter learning, and model selection be decoupled sequentially step by step. E.g., for the factorization by Eq. (23) we have

$$\begin{aligned}
&\min_{\text{a free } p(Y|X,\theta)} && KL(p(R|X)p(X)||q(X|R)q(R)) \\
&&& \Rightarrow \min KL(p(\theta|X)p(X)||q(X|\theta)q(\theta)), \\
&q(X|\theta) = && \int q(X|Y)q(Y)\mathrm{d}Y, \\
&\min_{\text{a free } p(\theta|X)} && KL(p(\theta|X)p(X)||q(X|\theta)q(\theta)) \\
&&& \Rightarrow \min KL(p(X)||q(X)), \\
&q(X) = && \int q(X|\theta)q(\theta)\mathrm{d}\theta. \qquad (49)
\end{aligned}$$

Also, as introduced in Sect. 5.1 of Ref. [1], Y may be divided into multiple layers and such a bottom-up decoupling nature still applies.

On one hand, this decoupling nature facilitates learning unknowns bottom-up layer by layer. On the other hand, learning within one layer becomes insensitive to the lower layers (especially their complexity). Consequently, it becomes poor on determining the complexity k of Y and its hierarchical configuration. Abandoning such a decoupling nature, the BYY best harmony learning makes automatic model selection become possible on each layer and each step.

4.2.3 Ying-Yang best harmony: Novelty and features

We further switch to the Box ④ in Fig. 4 to introduce Ying-Yang best harmony via maximizing $H_\mu(P||Q)$. Referring to Appendix A and Fig. A2 in Ref. [1], we start from observing how the special cases of maximizing $H_\mu(P||Q)$ leads to the following learning principles:

- Maximizing $H(p(Y|X)p(X)||q(X|Y)q(Y))$ with respect to a structure free $p(Y|X)$ leads to the maximum a posteriori (MAP) $\max_Y[q(X|Y)q(Y)]$, as shown by the Box ④a in Fig. 4. One example is competitive learning discussed around Eq. (4).
- Maximizing $H(p(\theta|X)p(X)||q(X|\theta)q(\theta))$ with respect to a free $p(\theta|X)$ leads to Bayes learning $\theta^* = \arg\max_\theta[q(X|\theta)q(\theta)]$, as shown in the Box ④b in Fig. 4. For a uniform or noninformative prior $q(\theta)$, it degenerates to the maximum likelihood again, sharing with Ying-Yang best matching.
- Maximizing $H(p(Y|X)p(X)||q(X|Y)q(Y))$ respect to a structure free $q(X|Y)$ leads to minimizing $KL(p(Y)||q(Y))$ with $p(Y) = \int p(Y|X)p(X)\mathrm{d}X$, as shown by the Box ④d in Fig. 4, from which we are again lead to MCE, MMI, INFORMAX in the Box ③d, sharing with Ying-Yang best matching again.
- In another special case $q(X|Y) = q(X)$, we have $H(p(Y|X)p(X)||q(X)q(Y)) = H(p(X)||q(X)) + H(p(Y)||q(Y))$. Maximizing $H(p(Y)||q(Y))$ with respect $q(Y)$ leads to $q(Y) = p(Y)$ and maximizing $H(p(Y)||p(Y))$ or equivalently minimum entropy, as shown by the Box ④c in Fig. 4. It is a counterpart of INFORMAX and thus called minimum information transfer (INFORMIN), covering those studies under the name of minor component or subspace analysis (MCA and MSA) [56,57], and further extensions to minor ICA (M-ICA) [47]. Readers are referred to a recent review [17].

In addition to the above degenerated cases, what is even important is that $H(p||q)$ or generally $H_\mu(P||Q)$ provides a favorable new learning principle when both $p(R|X)$ and $q(X|R)$ take parametric structures. The novelty of this principle can be observed from several different aspects, which are further addressed as follows.

First, Ying-Yang best harmony aims at a best Ying-Yang matching in a BYY system with a least complexity. Specifically, $\max_q H(p||q)$ for a fixed p forces the Ying machine $q(X|R)q(R)$ to best match $p(R|X)p(X)$. Due

to a finite size N and other existing constraints (if any), the limit $q(X|R)q(R) = p(R|X)p(X)$ may not be really reached. Still there is a trend towards this equality by which $H(p||q)$ (as illustrated by the Box ③c in Fig. 5), becomes the negative entropy, and its further maximization will minimize the system complexity, which makes the Ying-Yang pair in a least complexity.

Second, the least complexity nature may also be understood from the nature that $q(Y)$ is in a scale sensitive position within $H_\mu(p(R|X)p(X)||q(X|R)q(R))$ via $q(R) = q(Y|\theta_Y)q(\theta)$. More specifically, $H(p||q)$ contains one term that increases monotonically with $\int p(Y)\ln q(Y)\mathrm{d}Y$ that tends to $\int p(Y)\ln p(Y)\mathrm{d}Y$, for which a best harmony prefers one $q(Y)$ with a least entropy.

Generally, we say $q(Y)$ in a scale sensitive position within a cost function $\mathcal{F}(\,\cdot\,, q(Y))$ if this $\mathcal{F}(\,\cdot\,, q(Y))$ varies monotonically as the scale/complexity of $q(Y)$ varies such that maximizing or minimizing $\mathcal{F}(\,\cdot\,, q(Y))$ pushes this $q(Y)$ towards a least complexity. On the contrary, $q(Y)$ is not in such a scale sensitive position when $\mathcal{F}(\,\cdot\,, q(Y))$ is a likelihood function $\ln q(X|\theta) = \ln \int q(X|Y, \theta_{X|Y})q(Y|\theta_Y)\mathrm{d}Y$. Observing the example of the factor analysis by Eqs. (25) and (26), we have $q(X|\theta) = G(x|\mu, \Sigma)$ via $\Sigma = \sigma^2 I + A\Lambda A^{\mathrm{T}}$, where Σ is insensitive to the dimension of y. Also, $q(Y)$ is not in such a scale sensitive position when $\mathcal{F}(\,\cdot\,, q(Y))$ is given by $KL(p(Y|X)p(X)||q(X|Y)q(Y))$, which can be observed from the previous discussion that minimizing $KL(p(Y|X)p(X)||q(X|Y)q(Y))$ with respect to a free structure $p(Y|X)$ actually becomes equivalent to maximizing $\ln q(X|\theta)$.

For $q(Y)$ in a scale sensitive position, as addressed in Sect. 4.1 and Fig. 5 of Ref. [1], we can observe the scale k_Y of Y. Also this k_Y is usually a primary part of the entire scale set k. For many typical learning problems, e.g., Gaussian mixture in Fig. 1 and factor analysis by Eqs. (25) and (26), the task of model selection is just determining this complexity k_Y. Therefore, we are actually provided with a favorable new mechanism for model selection (particularly automatic model selection). Readers are referred to Sect. 2.2 and Fig. 5 in Ref. [1] for further details.

In summary, $q(Y)$ takes a role of at least equal importance to $q(\theta)$ for model selection, which thus provides favorable improvements on both model selection criteria and automatic model selection.

Third, the novelty of $H_\mu(P||Q)$ is also observed from that $H_\mu(P||Q)$ differs from $KL_\mu(P||Q)$ in the following aspects:

- $H_\mu(P||Q)$ is a triple-relation while it follows from the Box ②a in Fig. 5 that $KL_\mu(P||Q)$ is a degenerated case at $\mathrm{d}\mu = \mathrm{d}P$ for measuring a bi-relation.
- It follows from the Box ①b and Box ①c in Fig. 5 that maximizing $H_\mu(P||Q)$ consists of not only minimizing $KL(P||Q)$ for a Ying-Yang best match but also minimizing the information $-H(P||P)$ that is transferred by Yang. That is, Ying and Yang seeks a best agreement in a most tacit manner via a least amount of information communication.
- The bottom-up decoupling nature by Eq. (49) makes $q(Y)$ have no contribution to model selection. Instead, we are lead to $q(X|k)$ and accordingly the Box ④c in Fig. 5, that is, BIC [11] or MDL [54]. On the contrary, $H_\mu(p(R|X)p(X)||q(X|R)q(R))$ considers $q(R) = q(Y)q(\theta)$ with the following features:
  - The roles of $q(Y)$ and $q(\theta)$ are observed by $p(R|X) = p(Y, \theta|X)$ per instance of Y and per instance of θ. Not only the complexity k is observed via $q(\theta)$, but also the complexity k_Y is observed via $q(Y)$, such that Ying-Yang best harmony is able to make automatic model selection on each layer.
  - Without the above decoupling nature, the maximization of $H_\mu(P||Q)$ with respect to $p(R|X)$ also makes $p(Y, \theta|X)$ more selective to harmonize $q(X|R)q(R)$ via the best inner representations.

Fourth, we further observe the novelty of $H_\mu(P||Q)$ from its following differences from MCE [8–10]:

- As previously addressed at the end of Sect. 4.1, the name MCE was ever confusingly used in the literature of signal processing and information theory to refer both $\max_p H(p||q)$ and $\min_p KL(p||q)$ with a fixed q. Actually, the MCE studies have been widely made on $\min_p KL(p||q)$ with a fixed q. In contrast, $\max_p H(p||q)$ leads to $p(x) = \delta(x - c)$ when q is fixed while p is free of constraint, which has been regarded as a useless degenerated case, with no further effort made along this direction.
- The above apparent useless singular nature becomes useful and important when p, q are given by a BYY system. Because $p = p(R|X)p(X)$ includes $p(X) = p_h(X|X_N)$ by Eq. (20), $\max_p H(p||q)$ for a fixed q can not push $p(R|X)p(X)$ to entirely becomes a δ distribution, but push $p(R|X)$ into a most compact form under the constraint by $p(X) = p_h(X|X_N)$ and also by some structure of $p(R|X)$ (if any). Moreover, $\max_q H(p||q)$ for a fixed p forces the Ying machine $q(X|R)q(R)$ to best match $p(R|X)p(X)$ and accordingly become more compact too.
- For those MCE studies [8–10], a set θ of unknown parameters in p_θ is estimated via $\min_\theta KL(p_\theta||q)$ with a fixed q subject to a set of known constraints on p, which needs task dependent efforts to get

these constraints manually. In a BYY system, the set X_N of samples is directly input to the system as $p(X) = p_h(X|X_N)$ by Eq. (20), which is a general formulation that learns from its environment automatically.

Fifth, the following two aspects also provides insights on the novelty of $H_\mu(P||Q)$:

- Another information-theoretic perspective of BYY harmony learning can be found in Sect. II(C), Sect. II(E), and Fig. 3 of Ref. [4], which provides a three-level encoding scheme for optimal communication, being different from both the conventional MDL and the bit back MDL [58].
- As discussed around Eq. (25) in Ref. [1] and also around Eq. (10) in Sect. 2.1.5 of this paper, the gradient flow $\nabla_\varphi H(p||q)$ modifies the updating flow of the M-step in the EM algorithm for the maximum likelihood learning and Bayesian learning such that the learning dynamics has a mechanism similar to RPCL learning [24,59] as previously introduced in Sect. 2.2. Similar to that $\Delta\pi_{\ell,t}$ in Eq. (10) has two equivalent choices, $\Delta\pi\,(X, Y)$ in Eq. (25) of Ref. [1] corresponds to choice 1) of $\Delta\pi_{\ell,t}$ in Eq. (10) that describes the top-down fitness $Y \to X$. Also, we have its bottom-up equivalence to choice 2) of $\Delta\pi_{\ell,t}$ as follows:

$$\begin{aligned}\Delta\pi(X,Y) &= -E_{Y|X}(\theta_{y|x}) \\ &\quad + \int p\left(Y|X,\theta_{y|x}\right) E_{Y|X}(\theta_{y|x})\mathrm{d}Y, \\ E_{Y|X}(\theta_{y|x}) &= -\ln p\left(Y|X,\theta_{y|x}\right). \end{aligned} \tag{50}$$

In other words, the correcting term $\Delta\pi(X, Y)$ to the updating flow of the M-step in the EM algorithm can be interpreted from both a top-down perspective and a bottom-up perspective.

4.3 Learning implementation: Apex approximation, manifold shrinking, and balanced operation

4.3.1 Hierarchical implementation and apex approximation

After designing a BYY system as discussed in Sect. 3.2, the task of learning is determining all the unknowns in the BYY system by maximizing $H_\mu(P||Q)$. For simplicity and without losing generality, we focus on $H(p||q)$ given by Eq. (1).

Generally speaking, the maximization seeks the optimal inner representation $R^* = \{Y^*, \theta^*, \Xi^*, k^*\}$, featured by a hierarchical implementation. First, $H(p||q)$ by Eq. (1) is a function of the complexity k and also the hyper-parameter set Ξ if a prior $q(\theta)$ in Eq. (21) and its counterpart $p(\theta|X)$ in Eq. (23) contains hyper-parameters Ξ. Accordingly, we denote $H(p||q)$ by $H(k, \Xi)$, namely an objective function with respect discrete variables in k and continuous variables in Ξ. Its maximization needs two stages similar to Eq. (17), that is,

$$\begin{aligned} &\textbf{Stage I:} && \text{enumerating } k \text{ for a set of instances and} \\ &&& \text{getting } \Xi_k^* = \arg\max_\Xi H(k,\Xi), \\ &\textbf{Stage II:} && k^* = \arg\min_k J(k) = -H(k,\Xi_k^*). \end{aligned} \tag{51}$$

In fact, the function $H(k, \Xi)$ is not directly available, but needs to be computed from Eq. (1) via making three levels of integrals.

Moreover, we may replace X by X, h and put $p(X, h) = p_h(X|X_N)$ by Eq. (20) into Eq. (1), from which we have

$$H(p||q) = H_{h_0}(k, \Xi|X_N). \tag{52}$$

For simplicity, we start to consider the case with $h_0 = 0$, for which $p(X, h) = p(X) = \delta(X - X_N)$ and the above $H_{h_0}(k, \Xi|X_N)$ becomes

$$\begin{aligned} H(k,\Xi) &= H_0(k,\Xi|X_N) = H_{h_0=0}(k,\Xi|X_N) \\ &= \int p(Y,\theta|X_N)\pi(X_N,Y,\theta)\mathrm{d}Y\,\mathrm{d}\theta \\ &= \int p(\theta|X_N)H(p||q,\theta)\mathrm{d}\theta, \\ H(p||q,\theta) &= \int p(Y|X_N,\theta)\pi(X_N,Y,\theta)\mathrm{d}Y, \\ \pi(X_N,Y,\theta) &= \ln[q(X_N|Y,\theta)q(Y|\theta)q(\theta)], \end{aligned} \tag{53}$$

for which we need to handle the integrals of types $\int[\cdot]\mathrm{d}\theta$ and $\int[\cdot]\mathrm{d}Y$. The integrals sum up all the evidences for each possible scenario that Y, θ may take. For some tasks, the integrals are analytically trackable and thus solved manually, and then this problem reduces to just handling Eq. (51).

In order to avoid the difficulty of handling integrals over Y, θ, we start from considering the following optimal values:

$$\begin{aligned} Y^* &= \arg\max_Y \pi(X_N, Y, \theta), \\ \theta^* &= \arg\max_\theta H(p||q,\theta), \end{aligned} \tag{54}$$

and further pursuit along one of two typical directions.

One is approximately considering the integrals

$$\int_{\theta\in D^\rho_{\theta^*}} [\cdot]\mathrm{d}\theta, \qquad \int_{Y\in D^\rho_{Y^*}} [\cdot]\mathrm{d}Y,$$

within apex zones $D^\rho_{\theta^*}$ by Eq. (32) and $D^\rho_{Y^*}$ by Eq. (42).

When all the elements in Y are discrete, this apex approximation is relatively easy to handle. One simple example is J_t^κ in Eq. (14), and the other example is $C_t^\kappa(x_t)$ by Eq. (20) in Ref. [1] for BFA, where each element y in Y is a binary vector and thus C_t^κ consists of those ones that differ from y^* by one bit. For θ and also Y that takes real values, we approximately consider $D^\rho_{\theta^*}$ and $D^\rho_{Y^*}$ as hyper-spheres in some radius.

Another direction is considering the integrals centered around Y^*, θ^* with help of the following Taylor expansion around u^* up to the second order:

$$\begin{aligned}
&\int p(u)\,Q(u)\,\mathrm{d}u \approx Q(u^*) - \frac{1}{2}\mathrm{Tr}[\varepsilon_u\varepsilon_u^{\mathrm{T}}\Pi(u^*) + \Gamma_{\Pi}^{u}],\\
&u^* = \arg\max_u Q(u),\ \varepsilon_u = \eta_u - u^*,\\
&\Gamma_{\Pi}^{u} = \Gamma^u\Pi(u^*),\ \Pi(u) = -\frac{\partial^2 Q(u)}{\partial u \partial u^{\mathrm{T}}},
\end{aligned} \tag{55}$$

where η_u, Γ^u are the mean and the covariance of $p(u)$. This Taylor expansion was firstly used in Eq. (19) of Ref. [4], which is modified from a variant that makes Taylor expansion around the mean η_u up to the second order:

$$\int p(u)\,Q(u)\,\mathrm{d}u \approx Q(\eta_u) - \frac{1}{2}\mathrm{Tr}[\Gamma_{\Pi}^{u}], \tag{56}$$

which was previously used in Sect. 2.4 of Ref. [60] and Eq. (18) of Ref. [61].

Using Eq. (55) on the integral in Eq. (53), we get

$$\begin{aligned}
H(p||q,\theta) &= \pi(X_N, Y^*, \theta) - \frac{1}{2}\mathrm{Tr}[\varepsilon_Y\varepsilon_Y^{\mathrm{T}}\Pi_{Y|X,\theta}^{q} + \Gamma_{\Pi}^{Y|X}],\\
\varepsilon_Y &= \mathrm{vec}[Y^* - \eta_Y(X_N)],\\
\Gamma_{\Pi}^{Y|X} &= \Gamma_{Y|X,\theta}^{p}\Pi_{Y|X,\theta}^{q},\\
H(p||q) &= H(p||q,\theta^*) - \frac{1}{2}\mathrm{Tr}[\Gamma_{\Pi}^{\theta}],\\
\Gamma_{\Pi}^{\theta} &= \Gamma_{\theta|X}^{p}\Pi_{\theta|X}^{H},\\
\Pi_{\theta|X}^{H} &= -\frac{\partial^2 H(p||q,\theta)}{\partial\mathrm{vec}[\theta]\partial\mathrm{vec}[\theta]^{\mathrm{T}}} = \Pi_{\theta|Y,X}^{q} + \frac{1}{2}\Pi_{\theta}^{\Delta},\\
\Pi_{\theta}^{\Delta} &= \frac{\partial^2\mathrm{Tr}[\varepsilon_Y\varepsilon_Y^{\mathrm{T}}\Pi_{Y|X,\theta}^{q} + \Gamma_{\Pi}^{Y|X}]}{\partial\mathrm{vec}[\theta]\partial\mathrm{vec}[\theta]^{\mathrm{T}}},
\end{aligned} \tag{57}$$

where $\Pi_{\theta|Y,X}^{q}$ is given by Eq. (43), and $\Gamma_{Y|X,\theta}^{p}, \Gamma_{\theta|X}^{p}$, $\Pi_{Y|X,\theta}^{q}$ are given by Eq. (39), as well as $\eta_Y(X), \eta_\theta(X_N)$ are defined by Eq. (35). Specifically, we have $\eta_Y(X) = \eta_Y(X, \Phi)$ given by Eq. (37).

Hierarchically, $H(p||q)$ by Eq. (1) could be maximized in a multi-stage alternation as shown in Eq. (18). First, its Step Y gets Y^* by Eq. (54) and then removes the integral over Y to approximately get $H(p||q,\theta)$ by Eq. (57). Second, Step θ gets θ^* by Eq. (54) and then remove the integral over θ to get $H(p||q)$ by Eq. (57). Next, Step Ξ (also Stage I in Eq. (51)) gets Ξ^* by Eq. (57). Finally, Step k (also Stage II in Eq. (51)) selects one best k^* with the obtained Ξ^*. These steps are iteratively implemented and at each step we update one type of variables with the rest types fixed at their newest available values.

Generally, the case by Eq. (52) with $h \neq 0$ leads to

$$\begin{aligned}
H(p||q) &= H_{h_0}(k, \Xi|X_N)\\
&= H(p||q,\theta,X_N,h_0) + H(h_0),\\
H(h_0) &= \int p(h|h_0)\ln q(h|X_N)\mathrm{d}h,\\
H(p||q,\theta,X_N,h_0) &= \int p(h|h_0)H(p||q,\theta,X_N,h)\mathrm{d}h,\\
H(p||q,\theta,X_N,h) &= \int p_h(X|X_N)H_0(k,\Xi|X)\mathrm{d}X.
\end{aligned} \tag{58}$$

In the above $H(p||q,\theta,X_N,h)$, $p_h(X|X_N)$ takes the role of $p(u)$ and $H_0(k,\Xi|X)$ takes the role of $Q(u)$, it follows from Eq. (56) that

$$\begin{aligned}
H(p||q,\theta,X_N,h_0) &= H_0(k,\Xi|X_N) - \frac{1}{2}h_0^2\mathrm{Tr}[\Pi_X],\\
\Pi_X &= -\frac{\partial^2 H_0(k,\Xi|X)}{\partial\mathrm{vec}[X]\partial\mathrm{vec}[X]^{\mathrm{T}}}.
\end{aligned} \tag{59}$$

With h_0^2 given, the multi-stage alternation by Eq. (18) needs to be modified in Step θ, Step Ξ, and Step k with $H(p||q,\theta)$ replaced by the above $H(p||q,\theta,X_N,h_0)$. Moreover, we add the following step for updating h_0^2:

$$\begin{aligned}
&\text{Step } h_0: h_0^{2*} = \arg\max_{h_0}\{H(h_0) - \frac{1}{2}h_0^2\mathrm{Tr}[\Pi_X]\},\\
&\text{where}\quad q(h|X_N) \propto \begin{cases} 1/\sum_{t=1}^{N} p_h(x_t), & \text{(a)}\\ 1/p(X_N|X_N,h), & \text{(b)}\end{cases}
\end{aligned} \tag{60}$$

where $p_h(x)$ and $p(X|X_N,h)$ are given by Eq. (20), and $p(h|h_0)$ could be an exponential distribution with its mean h_0 or a Gamma distribution with a parameter h_0.

4.3.2 Manifold shrink: Automatic model selection

Typically, a learning process for θ^* is a process of optimizing an objective function with respect θ, with solution obtained as the objective function reaches its maximum ($< \infty$) or minimum value ($> -\infty$). Learning dynamics is finally stabilized or converged both to the value of unknown variables and to the value of the objective function.

However, the learning process for θ^* by maximizing $H(p||q)$ is quite different, which may consist of several sub-processes and each sub-process is featured by $H(p||q) \to \infty$ as one or a part of elements in θ tends zero (or a particular value).

Without losing generality, we consider the samples of $X_N = \{x_t\}$ that are independent and identically distributed (i.i.d.). In such cases, $H(p||q,\theta)$ in Eq. (53) and Eq. (57) is simplified into

$$\begin{aligned}
H(p||q,\theta) &= \sum_t H_t(p||q,\theta),\\
H_t(p||q,\theta) &= \int p(y|x_t)\pi(x_t,y,\theta)\mathrm{d}y\\
&\approx \pi(x_t,y_t^*,\theta) - \frac{1}{2}\mathrm{Tr}[\varepsilon_t\varepsilon_t^{\mathrm{T}}\Pi_{y|x,\theta}^{q} + \Gamma_{\Pi}^{y|x}],\\
\pi(x_t,y,\theta) &= \ln[q(x_t|y,\theta_{x|y})q(y|\theta_y)q(\theta)],\\
y_t^* &= \arg\max_y \pi(x_t,y,\theta),\\
\Gamma_{\Pi}^{y|x} &= \Gamma_{y|x,\theta}^{p}\Pi_{y|x,\theta}^{q},\\
\varepsilon_t &= y_t^* - \eta_y(x_t),
\end{aligned} \tag{61}$$

where $\Gamma_{y|x,\theta}^{p}$, $\Pi_{y|x,\theta}^{q}$, and $\eta_y(x_t)$ are given in Table 2 for Type S, which covers both having only x_t available and

knowing a given input-output pair $x_t \to y_t$ with help of simply $\eta_y(x_t)$.

Taking the factor analysis by Eqs. (25) and (26) as an example, we have

$$\pi(x_t, y, \theta) = \ln\left[G\left(x_t \,|Ay + \mu, \Sigma\right) G(y|\nu, \Lambda) q(\theta)\right], \\ \Pi^{y|x} = A^{\mathrm{T}}\Sigma^{-1}A + \Lambda^{-1}, \tag{62}$$

and the approximation $\approx$ in Eq. (61) becomes exactly $=$. Readers are referred to Eq. (17) in Ref. [1] for a detailed expression of $H(p||q, \theta)$ in Eq. (53).

Here, $\Lambda = \mathrm{diag}[\lambda_1, \lambda_2, \ldots, \lambda_m]$ takes the role of the SR parameters θ_{SR} as previously introduced in Sect. 2.2, with $\lambda_j = 0$ indicating that the jth dimension $y_t^{(j)}$ is extra. It follows from observing $\ln G(y|\nu, \Lambda)$ in $\pi_t(\theta, y)$ that $\lambda_j = 0$ is actually a singular point that contributes one ∞ to $H(p||q, \theta)$. When the sample variance of $y_t^{(j)}$ is not zero, we observe $\mathrm{d}H(p||q, \theta)/\mathrm{d}\lambda_j \to +\infty$ such that the singular point $\lambda_j = 0$ is unstable and the learning dynamics will push λ_j away from zero. When the sample variance of $y_t^{(j)}$ becomes zero, we have $H(p||q, \theta)/\mathrm{d}\lambda_j \to 0$ while $\mathrm{d}^2H(p||q, \theta)/\mathrm{d}^2\lambda_j \to -\infty$ as $\lambda_j \to 0$ such that $\lambda_j = 0$ and $H(p||q, \theta) \to \infty$ is a stable trap of "black hole" like for the learning dynamics.

Due to this particular nature of singularity, a learning dynamics starts from a manifold with a large enough dimension m and evolves as illustrated roughly in Fig. 6. When one inner dimension $y_t^{(j)}$ is extra, the value of $y_t^{(j)}$ will take a zero or constant, and the sample variance of $y_t^{(j)}$ becomes zero, there will be one or more "black hole" like traps that capture learning dynamics. Once falling into such a trap, learning dynamics on $H(p||q, \theta)$ is buried because $H(p||q, \theta) \to \infty$. One trick is to remove this trap by simply discarding the variable λ_j and the corresponding part to restore learning dynamics.

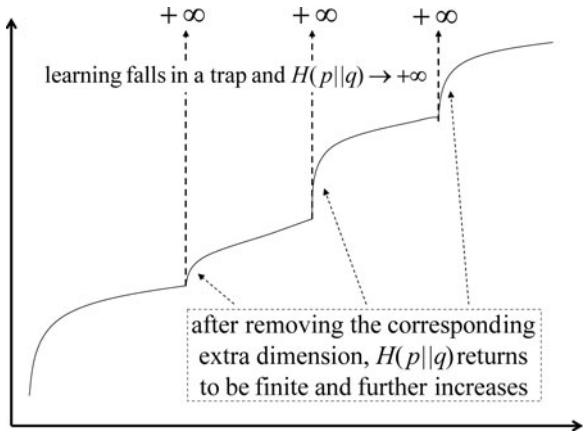

Fig. 6 Learning dynamics with black hole like traps

As a whole, the learning process proceeds as illustrated in Fig. 6 until all the extra dimensions of y have been removed. In other words, automatic model selection happens during this learning process. Finally, the learning dynamics is stabilized or converged to the maximum of $H(p||q, \theta)$ ($< \infty$) in a way similar to one standard optimization process.

For some learning tasks, there may be apparently no such a "black hole" like trap. For Gaussian mixture in Fig. 1, the harmony functional by Eq. (1) gets a detailed expression given by Eq. (10) in Ref. [1]. Discarding the data smoothing part by letting $h = 0$, we have

$$H(p||q, \theta) = \sum_{t,j} q(j|x_t, \theta) \ln[G(x_t|\mu_j, \Sigma_j)\alpha_j q(\theta)], \tag{63}$$

where $\alpha_1, \alpha_2, \ldots, \alpha_k$ takes the role of the SR parameters θ_{SR} as previously introduced in Sect. 2.2, with $\alpha_j = 0$ indicating that the jth Gaussian component is extra. As $\alpha_j \to 0$, we have $\sum_t q(j|x_t, \theta) \ln \alpha_j \to N\alpha_j \ln \alpha_j \to 0$, which will not cause $H(p||q, \theta) \to \infty$. In other words, $\alpha_j = 0$ is apparently not a singular point of $H(p||q, \theta)$, while this learning process is similar to a standard optimization process. Actually, this scenario could be understood by observing that the density $q(Y)$ is replaced by a discrete probability $q(y = j) = \alpha_j$ with an infinite $\ln \delta(y = j)$ discarded already before we get Eq. (62).

Learning dynamics of maximizing $H(p||q, \theta)$ may be roughly depicted as a process of manifold shrinking. Starting from one point on a high dimension manifold, learning searches within this manifold and may be captured by a "black hole" like trap with an infinite energy or potential. Releasing such an infinite energy makes the manifold collapse or shrink into a lower dimension. After a number of such manifold shrinking, automatic model selection is achieved, and learning dynamics finally proceeds within one manifold with a stabled dimension and eventually stabilizes or converges to one point with a maximum of $H(p||q, \theta)$.

4.3.3 Balanced operation and computing order

Another insight may come from taking Eq. (62) as an example. The corresponding $H(p||q, \theta)$ contains a term $-0.5\mathrm{Tr}[\Gamma_{\Pi}^{y|x}]$ with $\Gamma_{\Pi}^{y|x} = \Gamma^{y|x}\Pi^{y|x}$. It follows from the variety preservation principle by Eq. (39) that one choice is $\Gamma^{y|x} = \Pi^{y|x\,-1}$ and thus $\Gamma_{\Pi}^{y|x} = I$. The term $-0.5\mathrm{Tr}[\Gamma_{\Pi}^{y|x}] = -0.5m$ becomes not differentiable. Though it still contributes to Step k in Eq. (18) as we use a model selection criterion for selecting m, not only it has no help on learning dynamics for automatic model selection, but also it makes learning prone to a local maximum of $H(p||q, \theta)$. One remedy is letting $\Gamma^{y|x} = \Pi^{y|x\,-1} + \rho$ such that $\Gamma_{\Pi}^{y|x} = I + \rho\Pi^{y|x}$ with a diagonal matrix $\rho \to 0$ gradually, e.g., see Eqs. (17) and (40) in Ref. [1].

On the other hand, maximizing $H(p||q, \theta)$ by Eq. (57) with respect to $\mu(x, W)$ as a free vector results in $\varepsilon_t = 0$ and $\frac{1}{2}\mathrm{Tr}[\varepsilon_t\varepsilon_t^{\mathrm{T}}\Pi^{y|x}] = \frac{1}{2}\varepsilon_t^{\mathrm{T}}\Pi^{y|x}\varepsilon_t = 0$ which also makes

learning quench to a local maximum of $H(p||q)$. Instead, learning is made via minimizing $\varepsilon_t^{\mathrm{T}}\Pi^{y|x}\varepsilon_t$ with respect to W, which makes $\varepsilon_t \to 0$ gradually.

In general, we encounter a similar scenario as we consider the term $\mathrm{Tr}[\Gamma_{\Pi}^{Y|X}]$ in Eq. (57). Also, $\eta_Y(X) = \eta_Y(X,\Phi)$ by Eq. (37) takes a role similar to the above $\mu(x,W)$.

For the integral over θ, there is also a term $\mathrm{Tr}[\Gamma_{\Pi}^{\theta}]$ in Eq. (57). But $\Gamma_{\theta|X}^{p}$ by Eq. (39) is actually different from $\Pi_{\theta|X}^{H}$. From $\ln\int q(X_N|Y,\theta)q(Y|\theta)\mathrm{d}Y = 0.5d_Y\ln(2\pi) + \ln[q(X_N|Y^*,\theta)q(Y^*|\theta)] - 0.5\ln|\Pi_{Y|X,\theta}^{q}|$, it follows that

$$\Pi_{\theta|X}^{q} = \Pi_{\theta|Y,X}^{q} + \frac{1}{2}\Pi_{\theta}^{\mathrm{Det}},$$
$$\Pi_{\theta}^{\mathrm{Det}} = \frac{\ln|\Pi_{Y|X,\theta}^{q}|}{\partial\mathrm{vec}[\theta]\partial\mathrm{vec}[\theta]^{\mathrm{T}}}.$$

From $\Gamma_{\theta|X}^{p} = \Gamma_{\theta|X}^{q} = \Pi_{\theta|X}^{q\ -1}$ in Eq. (39), even when we get $\Pi_{\theta}^{\Delta} = 0$ as $\varepsilon_Y = 0$ and $\Gamma_{\Pi}^{Y|X} = I$, we still have $\Gamma_{\Pi}^{\theta} = \Gamma_{\theta|X}^{p}\Pi_{\theta|X}^{H} = [\Pi_{\theta|Y,X}^{q} + \frac{1}{2}\Pi_{\theta}^{\mathrm{Det}}]^{-1}[\Pi_{\theta|Y,X}^{q} + \frac{1}{2}\Pi_{\theta}^{\Delta}] \neq I$.

In other words, the term $\mathrm{Tr}[\Gamma_{\Pi}^{\theta}]$ in Eq. (57) does contribute to $H(p||q)$ for updating Ξ^* in Eq. (51) or Eq. (18), which is one development from the previously studies, e.g., as introduced in Fig. 5 and Eq. (36) in Ref. [1], where it was roughly assumed that $\Gamma_{\theta|X}^{p} \approx \Pi_{\theta|Y,X}^{q\ -1}$ with $\Pi_{\theta}^{\mathrm{Det}}$ ignored. That is, the term $\mathrm{Tr}[\Gamma_{\Pi}^{\theta}] = d_\theta$ is the number of free parameters in θ, which contributes to Step k in Eq. (18), but has no help to updating Ξ.

To make the term $\mathrm{Tr}[\Gamma_{\Pi}^{\theta}]$ helpful to updating Ξ, one other way is letting $\Gamma_{\theta|X}^{p} \approx \Pi_{\theta|Y,X}^{q\ -1} + \rho$ and controlling a diagonal matrix $\rho \to 0$ gradually. Also, another way is letting $\Theta^{\mu}(\Xi)$ to be the value of Θ^* at a past time such that $\Theta^{\mu}(\Xi) - \Theta^* \to 0$ as learning proceeds, e.g., Eq. (33) and Fig. 5 in Ref. [62].

Instead of Eq. (58), it follows from the factorization of $p(Y,\theta|X) = p(Y|X)p(\theta|Y,X)$ by Eq. (41) that we can alternatively rewrite Eq. (53) into

$$\begin{aligned}
H(k,\Xi) &= H_0(k,\Xi|X_N) = H_{h_0=0}(k,\Xi|X_N)\\
&= \int p(Y|X_N)p(\theta|Y,X_N)\pi(X_N,Y,\theta)\mathrm{d}Y\mathrm{d}\theta\\
&= \int p(Y|X_N)H(p||q,Y)\mathrm{d}Y,\\
H(p||q,Y) &= \int p(\theta|Y,X_N)\pi(X_N,Y,\theta)\mathrm{d}\theta,\\
\pi(X_N,Y,\theta) &= \ln[q(X_N|Y,\theta)q(Y|\theta)q(\theta)].
\end{aligned} \tag{64}$$

One choice is considering $q(\theta)$ and $p(\theta|Y,X_N)$ in a conjugated pair such that the integral over θ or the integrals over both θ and Y can be analytically solved, e.g., considering the DNW conjugated pair for Gaussian mixture in Ref. [7].

The other choice is removing the integral over Y approximately as follows:

$$H(p||q) = H(p||q,Y^*) - \frac{1}{2}\mathrm{Tr}[\varepsilon_Y\varepsilon_Y^{\mathrm{T}}\Pi_{Y|X}^{\mathrm{H}} + \Gamma_{\Pi}^{Y|X}],$$

$$\begin{aligned}
\varepsilon_Y &= \mathrm{vec}[Y^* - \eta_Y(X_N)],\\
\eta_Y(X_N) &= E_{p(Y|X_N)}(Y),\\
Y^* &= \arg\max_{Y} H(p||q,Y),\\
\Gamma_{\Pi}^{Y|X} &= \Gamma_{Y|X}^{p}\Pi_{Y|X}^{\mathrm{H}},\ \Gamma_{Y|X}^{p} = \Pi_{Y|X}^{q\ -1},\\
\Pi_{Y|X}^{\mathrm{H}} &= -\frac{\partial^2 H(p||q,Y)}{\partial\mathrm{vec}[Y]\partial\mathrm{vec}[Y]^{\mathrm{T}}},
\end{aligned} \tag{65}$$

where $\Pi_{Y|X}^{q}$ is given by Eq. (43). Approximately, we have

$$\begin{aligned}
&\int q\left(X|R\right)q\left(R\right)\mathrm{d}\theta\\
&= \int q(X_N|Y,\theta)q(Y|\theta)q(\theta)\mathrm{d}\theta\\
&= \pi(X_N,Y,\theta^*) - 0.5\ln|\Pi_{\theta|Y,X}^{q}| + 0.5d_Y\ln(2\pi),\\
\theta^* &= \arg\max_{\theta}\pi(X_N,Y,\theta).
\end{aligned} \tag{66}$$

It further follows from $\Pi_{Y|X}^{q}$ in Eq. (43) that we have

$$\Pi_{Y|X}^{q} = \Pi_{Y|X,\theta}^{q} + \frac{1}{2}\Pi_{Y}^{\mathrm{Det}},$$
$$\Pi_{Y}^{\mathrm{Det}} = \frac{\ln|\Pi_{\theta|Y,X}^{q}|}{\partial\mathrm{vec}[Y]\partial\mathrm{vec}[Y]^{\mathrm{T}}}. \tag{67}$$

The third choice is getting $H(p||q,Y)$ by removing the integral over Y approximately as follows:

$$H(p||q,Y) = \pi(X_N,Y,\theta^*) - \frac{1}{2}d_\theta, \tag{68}$$

where $d_\theta = \mathrm{Tr}[\Gamma_{\Pi}^{\theta|Y,X}]$ and it follows from Eq. (43) that $\Gamma_{\Pi}^{\theta|Y,X} = \Gamma_{\theta|Y,X}^{p}\Pi_{\theta|Y,X}^{q} = I$. Putting the above $H(p||q,Y)$ into Eq. (65), we simply get

$$\Pi_{Y|X}^{H} = -\frac{\partial^2 H(p||q,Y)}{\partial\mathrm{vec}[Y]\partial\mathrm{vec}[Y]^{\mathrm{T}}} = \Pi_{Y|X,\theta}^{q}. \tag{69}$$

For the general case by Eq. (52) with $h \neq 0$, we need to add one term $0.5h_0^2\mathrm{Tr}[\Pi_X]$ in Eq. (59) and update h by Eq. (60).

Additionally, we consider a partition $\theta = \theta_a \cup \theta_b, \theta_a \cap \theta_b = \emptyset$ with $q(\theta) = q(\theta_a)q(\theta_b|\Xi^q)$, i.e., one part has hyper-parameters while the other part has no hyper-parameters (e.g., Jeffreys priors). Accordingly, we have

$$\begin{aligned}
H_0(k,\Xi|X_N) &= Part_a + H_b(\Xi),\\
H_b(\Xi) &= \int p(\theta_b|X_N,\Xi^p)\ln q(\theta_b|\Xi^q)\mathrm{d}\theta_b,
\end{aligned} \tag{70}$$

where $\Xi = \{\Xi^p, \Xi^q\}$. We handle $Part_a$ in a way same as $H_0(k,\Xi|X_N)$, plus the contribution by $H_b(\Xi)$.

4.4 Historical remarks and demanding topics

4.4.1 Historical remarks

Jointly considering the Yang machine and Ying machine in a Bayesian Ying-Yang system, the learning principle is featured by a measure for bi-entity proximity between the probabilistic structures of $p(R|X)p(X)$ and

$q(X|R)q(R)$. Studies on both Ying-Yang best matching by minimizing $KL(p||q)$ and Ying-Yang best harmony by maximizing $H(p||q)$ were initialized in 1995 [2].

On one hand, the KL divergence, as shown by the Box ②b in Fig. 5, was widely used to measure the discrepancy between two distributions in the areas of information theoretical approaches, and also brought to the areas of machine learning as one popular learning principle. Naturally, this KL divergence was adopted for measuring Ying-Yang matching in Ref. [2]. Actually, one focus of the early period of BYY learning studies is exploring how minimizing the KL divergence between $p(Y|X)p(X)$ and $q(X|Y)q(Y)$, with four components $q(X|Y)$, $q(Y)$, $p(Y|X)$, and $p(X)$ in different structures, leads to a number of existing approaches of unsupervised learning and supervised learning [2,46,63–66], under the names of Bayesian-Kullback Ying-Yang (BKYY) learning/machine or BYY KL learning, etc.

Moreover, extension has also been suggested from KL divergence based BYY matching to non-KL divergence based BYY matching with $\ln r$ extended to a general convex function [67]. Also, under the name of Bayesian Convex Ying-Yang (BCYY) learning, a re-weighted EM (REM) algorithm is developed in Ref. [68] for Gaussian mixture, which is empirically shown to be more robust to outliers. Readers are referred to Sects. 22.9.1, 22.9.2 and 22.6.3 of Ref. [42] for detailed historical remarks on studies made before 2003, and also to Appendix A of Ref. [1] and Sect. 4.2.2 of this paper for additional efforts.

On the other hand, explorations on Ying-Yang best harmony by maximizing $H(p||q)$ also started from the above early period of the BYY learning studies. Particularly, efforts were made on Gaussian mixture in Sects. 4, 5, and 6 of Ref. [2], which has actually started the following threads of studies on the BYY best harmony learning:

- The hard-cut EM algorithm in Table 1 (or WTA-BYY harmony in Fig. 7 of Ref. [1]) was firstly proposed in Sect. 4.2 of Ref. [2], where Eqs. (19) and (20) are equivalent to the following simplified version of $H(p||q)$:

$$H(p\|q) = \sum_t \sum_\ell p(\ell|x_t,\theta) \ln G(x_t|\mu_\ell, \Sigma_\ell) + \sum_\ell \alpha_\ell \ln \alpha_\ell,$$
$$p(\ell|x_t,\theta) = \delta_{\ell,\ell^*(x_t)} \text{ is given by Eq. (6)}, \quad (71)$$

  which comes naturally via maximizing $H(p||q)$ by Eq. (11) with respect a free $p(j|x_t,\theta)$. In Sect. 4.2 of Ref. [2], Eq. (71) came from the corresponding KL divergence $KL(p||q)$ by heuristically imposing the constraint $p(\ell|x_t,\theta) = \delta_{\ell,\ell^*(x_t)}$, motivated by the winer-take-all competition used in the classic minimum mean square error clustering or equivalently vector quantization. It is this motivation that leads us from considering $KL(p||q)$ to move into considering the above special case of $H(p||q)$ by Eq. (71).
- Actually, such a link between $KL(p||q)$ and $H(p||q)$ is an example of putting certain constraints to the general relation between $KL(p||q)$ and $H(p||q)$ as shown by the Box ①c in Fig. 5. This general relation was also firstly studied in the notation $KL(p||q) = H - Q + D$ by Eqs. (8), (11), and (12) in Ref. [2]. A more general constrained linkage is further presented by Eq. (45) in Ref. [4].
- Applied to the classic minimum mean square error (MSE) clustering, Eq. (20) in Ref. [2] is further simplified into a criterion $J(k)$ by Eq. (24) in Ref. [2] for selecting the number of clusters, which is actually the first example of new model selection criteria obtained from $H(p||q)$. Subsequently in 1996 [64,69], this $J(k)$ was experimentally verified and further extended to supervised learning, and then generalized and investigated both theoretically and experimentally in 1997 [68].
- The basic idea of BYY harmony learning based automatic model selection was also firstly presented in Sect. 5.2 of Ref. [2]. For the two features of automatic model selection addressed in Sect. 2.2.2, the first one (e.g., $\alpha_l \to 0$ and $\Sigma_l \to 0$) was basically addressed, while the second one was partially discussed via a special observation that the first feature emerges during learning by the above mentioned hard-cut EM algorithm or WTA-BYY harmony learning.
- Also, a preliminary effort was made on building up a link between RPCL learning and the BYY harmony learning in Sect. 6.2 of Ref. [2].

Started from 1997, a general expression of harmony measure $H(p||q) = \int p(Y|X)p(X) \ln[q(X|Y)q(Y)]\mathrm{d}X\mathrm{d}Y$ is suggested as a general model selection criterion under the notation $J_2(k)$, see Eqs. (3.8) and (3.9) in Ref. [26], Eqs. (13) and (15) in Ref. [63], and Eq. (12) in Ref. [68]. Also, this general criterion has been applied to several typical learning models. Readers are referred to historical remarks given in Sect. 23.7.1 of Ref. [42] for studies made before 2003, and also to Table 3 in Ref. [35], Table 3 in Ref. [47], Fig. 2 in Ref. [4], Sect. 3.4.4 in Ref. [41], as well as Eqs. (42)–(45), (60), and (65) in Ref. [5] for additional efforts.

Since 1999, studies have proceeded to using this general form $H(p||q)$ for both parameter learning and model selection, see Eqs. (3) and (4) in Ref. [70], Eqs. (5) and (6) in Ref. [71], Eqs. (5) and (8) in Ref. [60], as well as Eqs. (8) and (10) in Ref. [61], including parameter learning with automated model selection (see Eqs. (28) and (29) in Ref. [39]). In the subsequent decade, extensive efforts have been systematically conduced on the BYY best harmony learning [4,5,25,35,38,40,41,47,48,72], covering not only theoretical analysis and deeper

understanding on fundamental issues of this learning principle and its relations to other existing typical learning principles and approaches, but also developments of system design principles and implementing techniques, as well as learning algorithms for a number of typical learning tasks. Readers are referred to historical remarks given in Sects. 23.7.3 and 23.7.4 of Ref. [42] for studies made before 2003, and also to Ref. [1] and Sect .4.2.3 of this paper for recent reviews.

Both $KL(p||q)$ and $H(p||q)$ includes $p(X) = \delta(X - X_N)$ that comes directly from a sample set X_N, which incurs for the problem of a small size of samples when N is not large enough. In addition to select a model with an appropriate complexity, learning regularization tackles this problem by adding some constraint to make its model complexity reduced effectively. In a BYY system, learning regularization has been made with some constraint added to each of four components $q(X|Y)$, $q(Y)$, $p(Y|X)$, and $p(X)$ under the following notations:

- Data smoothing regularization: each sample is assumed to come from a local smooth structure such that $p(X)$ is constrained to be a mixture of local distributions with each centered at one sample, e.g., $p(X) = p_h(X|X_N)$ by Eq. (20). The idea started from Eq. (5) in Ref. [2] and Eq. (1) in Ref. [64]. In 1997, it was further named data smoothing (see Eq. (3.10) in Ref. [26]), with an appropriate h learned via the KL learning by Eq. (7) in Ref. [33]. Also, data smoothing is suggested in Ref. [63] for supervised learning of three-layer forward net and mixture of experts. Moreover, a smoothed EM algorithm is given for Gaussian mixture (see Eq. (18) in Ref. [63]). Additionally, the second order approximation by Eqs. (56) and (59) was further developed in Sect. 2.4 of Ref. [60].
- Normalization regularization: It follows from Eq. (31) that the constraint $\int q(u|\theta)\mathrm{d}u = 1$ breaks down on a finite set of samples $\{u_t\}$. To re-ensure constraint, in Refs. [39,40,61] we normalize $q(u|\theta)$ into

$$\tilde{q}(u|\theta) = q(x|\theta)/Z(\theta), \quad Z(\theta) = \sum_t q(u_t|\theta), \quad (72)$$

  which causes a conscience de-learning that not only introduces a regularization to the ML learning, but also makes the BYY harmony learning behave similar to the RPCL learning.
- Structural regularization: as addressed by Items (A), (B), and (C) in Sect. 3.4.2, an appropriate structure for $p(Y|X)$ actually provides a type of regularization, which was firstly suggested in 1997 (see Item 3.4 in Ref. [26]).

Also, regularization emerges effectively with $f(r) = \ln r$ by Eq. (47) replaced by a convex function $f(r)$, as demonstrated by experiments on Gaussian mixture [68]. Moreover, we may regard $KL(p||q)$ as a regularized version of $H(p||q)$, and then make learning gradually shift from minimizing $KL(p||q)$ to maximizing $H(p||q)$ with help of a simulated annealing procedure.

Readers are referred to Sect. 23.7.4 of Ref. [42] for historical remarks on these types of learning regularization, and also to Items (a) and (b) of the next subsection for further discussions. Additionally, these types of learning regularization also provided new variants and extensions to those KL-divergence based learning approaches, see Sect. 22.9.2 in Ref. [42].

4.4.2 Demanding topics

Similar to Sect. 3.4.2, we summarize a number of topics about both challenging problems and interesting issues for future efforts.

(a) As addressed in Sects. 4.1.1 and 4.1.2, one most important case of the harmony functional $H_\mu(P||Q)$ is featured with $f(r) = \ln r$ by Eq. (47). On one hand, $\max_p H(p||q)$ leads to $p(x) = \delta(x - c)$. This feature applies to any $f(r) \neq r$ that monotonically increases with r. On the other hand, $\max_q H(p||q)$ leads to $q = p$ (i.e., a feature of best matching) for $f(r) = \lambda \ln r + c$ with any constant c and a constant $\lambda > 0$. However, we no longer have $q = p$ when $f(r)$ is a general convex function $\mathrm{d}^2 f(r)/\mathrm{d}r^2 < 0$ [73]. E.g., maximizing $H(p\|q) = \sum_{t=1}^N p_t f(q_t)$ with respect to q results in

$$q_t = \frac{f'\left(\frac{1}{p_t}\right)}{\sum_{t=1}^N f'\left(\frac{1}{p_t}\right)}, \quad f'(r) = \mathrm{d}f(r)/\mathrm{d}r, \quad (73)$$

which was given by Eq. (83) in Ref. [47] and Eq. (22.49) in Ref. [42]. Moreover, $f(r)$ may be classified as super-ln (e.g., one is the so-called α-function) if its $f'(r)$ decreases with r in a rate slower than $1/r$, or otherwise as sub-ln (e.g., a negated α-function). Similar to Eq. (46), we get an alternative spectrum for further investigations. Different types of $f(r)$ lead to different relation between p, q, which indicates that these types of $f(r)$ need to be included into those topics in Sect. 3.4.2 in coordination with Yang structure design.

(b) It follows from the Box ①b of Fig. 5 that $H_\mu(P||Q)$ with $f(r) = \ln r$ has the following simple additive decomposition

$$\begin{aligned} H_\mu(P||Q) &= H_\mu(P||P) + H_P(P||Q) \\ &= H_\mu(P||P) - KL(P||Q). \end{aligned} \quad (74)$$

Alternatively, we may redefine $H_\mu(P||Q)$ by extending this nature to a general case that even $f(r)$ becomes super-ln or sub-ln, e.g., see Sect. 2 in Ref. [70], Sect. 2.2 in Refs. [71] and [74], as well as Sect. II(B) and Eq. (8) in Ref. [61]. We need to

further investigate whether this simple combination still works or otherwise how the two terms in Eq. (74) are better combined. One way is a linear combination by Eq. (23.49) in Ref. [42], Eq. (49) in Ref. [46], Eq. (8) in Ref. [66], and Eq. (22) in Ref. [65]. Investigations are needed not only on justifying whether such a linear combination works but also how weighting coefficients are appropriately determined. Further effort may also be put on a simulated annealing procedure that making learning gradually shift from minimizing $KL(p||q)$ to maximizing $H(p||q)$.

(c) $H_\mu(P||Q)$ in Fig. 5 or $H(p||q)$ by Eq. (1) is formulated with both p, q in density functions, which is appropriate when all the variables in R are real numbers. However, $H(p||q)$ by either of Eq. (11), Eq. (63), and Eq. (71) considers a discrete distribution function $q(y = j) = \alpha_j$, which comes from putting a density function $q(Y)$ that actually contains an infinite $\ln \delta(y-j)$ term into $H(p||q)$ by Eq. (1). Similar to Fig. 6, this infinite $\ln \delta(y-j)$ term is discarded by an external option, which leads to $H(p||q)$ by Eq. (11) and Eq. (63). Such an external treatment needs further theoretical justification. Also, such a density based $H(p||q)$ incurs the difficulty of handling the integrals. In a real implementation, this $H(p||q)$ can not be computed exactly but evaluated approximately on a set of discrete points of samples.

Alternatively, $H(p||q)$ may be re-defined with both p, q in discrete distribution functions. It avoids to handle integrals (see Eqs. (21) and (22) in Ref. [39] and Sect. II(A) of Ref. [40]). However, for a density function $q(u|\theta)$ of a continuous variable u, we need to turn it into a discrete distribution by Eq. (72) for being put into $H(p||q)$, which incurs approximations too. Usually, we have only a set of samples for X. It remains to be a challenge problem to get a set of samples about Y, θ to handle $p(Y|X, \theta)$, $p(\theta|X)$. Particularly, only turning the components of Ying machine q into discrete distributions by Eq. (72), we get a distribution based $H(p||q)$ that differs from its density based counterpart in merely one extra term $-\ln Z(\theta)$, which can be interpreted as a prior $q(\theta)$ as previously discussed around Eq. (31). In such a case, two formulations of $H(p||q)$ meet. Still, investigations are needed on exploring their further relations. Moreover, both of them need certain approximation to implement, which naturally rises a question to ask which formulation is better, and whether we can combine the advantages of each.

(d) Tackling the task of learning the regularity underlying a small size of samples X_N, the purpose of both model selection and learning regularization is controlling an appropriate model complexity for a better generalization performance, which is usually featured by the following two points:

- Stability: though the generalization performance of a learned model will deteriorate as N reduces, we desire that it deteriorates as slowly as possible. That is, we prefer that the generalization performance is as stable as possible.
- Optimality: we desire the generalization performance of a learned model is as close as possible to the best generalization performance.

However, it remains an open challenge on how the harmony measure $H(p||q)$ is related to the best generalization performance. Conceptually, knowing the best generalization performance needs examining all the samples that are out of X_N but share the same regularity underlying X_N. Thus, it is very difficult to evaluate such a best performance. Though a very rough bound may be theoretically estimated subject to certain impractical assumptions, whether we should directly target at this optimality is actually itself an open question too. Instead, further efforts are deserved on taking the above stability in consideration of maximizing the harmony measure $H(p||q)$, together with a systematical comparison on those efforts introduced in Sect. 4.4.1, such as data smoothing regularization, normalization regularization, and structural regularization.

(e) It is insightful to make a comparative investigation on those discriminative training criteria in Eq. (48), for which Eq. (1) is simplified as follows:

$$H_{\mathrm{BYY}}(Y_N, X_N, \theta) = \sum_{r, Y \in D_Y^{(r)}} p(Y|X^{(r)}, \theta) \ln [q(X^{(r)}|Y, \theta_{x|y}) q(Y|\theta_y)], \tag{75}$$

which is written in a same format as Eq. (48). It differs from $F_{\mathrm{MMI}}(Y_N, X_N, \theta)$ that is equivalent to maximizing the Yang passway $p(Y_N|X_N)$ and also differs from $F_{\mathrm{MCE}}(Y_N, X_N, \theta)$ [50–53] that is regarded as an extension of $F_{\mathrm{MMI}}(Y_N, X_N, \theta)$ with $\ln p(Y^{(r)}|X^{(r)}, \theta)$ replaced by $s(1 - \frac{1}{p(Y^{(r)}|X^{(r)}, \theta)})$. Though $F_{\mathrm{MCE}}(Y_N, X_N, \theta)$ conceptually shares with RPCL [24,59] the idea that enhances the winner and penalizes the rivals, the two are considerably different both in formulae and implementation. RPCL learning in Table 1 can be regarded as an approximate implementation of $H_{\mathrm{BYY}}(Y_N, X_N, \theta)$. Comparing $H_{\mathrm{BYY}}(Y_N, X_N, \theta)$ with $F_{\mathrm{FPE}}(Y_N, X_N, \theta)$, we observe that $\ln [q(X^{(r)}|Y^{(r)}, \theta_{x|y}) q(Y|\theta_y)]$ and $L(Y, Y^{(r)})$ locate at a same position but very different in their details, which leads to differences in their implementing algorithms and learning performances [75]. Still, it deserves a further comparative

investigation.

(f) Conventionally, a process of parameter learning on a parametric model $q(x|\theta)$ is formulated as a process of optimizing an objective function with respect to θ, with a learning dynamics finally stabilized or converged both to the value of unknown variables and to the value of the objective function. Further mathematical analysis on such type of learning processes can be made with help of information geometry theory [76]. However, the learning process for θ by maximizing $H(p||q)$ is very different, as outlined in Sect. 4.3.2 and illustrated in Fig. 6. Neither the conventional optimization theory nor the existing information geometry theory can be applied. Details about this learning dynamics need to be explored. Not only unclear points are waiting for clarification, but also a new information geometry theory is expected.

(g) As illustrated in Fig. 6, parameter learning and model selection is a joint process that consists of both continuous optimization and discrete optimization. A conventional two stage implementation decouples this joint process into an outer loop of discrete optimization and an inner loop of continuous optimization, which becomes conceptually implementable but practically costs extensive computations. Instead of being intrinsic, such a decoupling has actually removed all "black hole" like traps in Fig. 6 without justifications, which makes the nature of automatic model selection lost unfavorably. Motivated from Sect. 4.3.2, efforts are needed on how to handle the joint process while avoiding to make two types of optimization decoupled unfavorably. Observing one example in Sect. 4.3.3, $H(p||q,\theta)$ for the FA contains a term $-0.5\text{Tr}[\Gamma_\Pi^{y|x}]$ that degenerates to a not differentiable discrete number $-0.5m$ (like a quantized energy) when $\Gamma^{y|x} = \Pi^{y|x\ -1}$, i.e., it has been decoupled into a part of discrete optimization and thus has no help on learning dynamics for automatic model selection. Efforts are needed on how to interpret this quantizing phenomenon and how to avoid such types of unfavorable decoupling. E.g., the implementation by Eq. (64) differs from the implementation by Eq. (58) in that the term $\text{Tr}[\Gamma_\Pi^{Y|X}]$ is still functioning within $H(p||q)$ without becoming a discrete number.

(h) The implementing technique by Eq. (55) takes an important role in removing the integral over Y and the integral over θ. However, this implementation also leads to a term $0.5\text{Tr}[\Gamma_\Pi^u]$ that may becomes a discrete number. Alternatively, $p(Y|X,\theta)$ was suggested as either a mixture of several analytically solvable conditional distributions (e.g., Eq. (10) in Ref. [65]) or a mixture of experts (e.g., Eq. (14) in Ref. [39] and Eq. (22.21) in Ref. [42]), which needs a further study too. Moreover, we also need to seek some new optimization techniques for a maximization of $H(p||q)$.

5 Insights on inner dependence structures

From the least complexity principle introduced in Sect. 3.1 and especially Eqs. (21) and (22), the simplest structure of inner representation R is, in the cases without any given knowledge, that all elements in R are mutually independent and each element is featured by a probabilistic structure, while the dependence structures among the observed samples in X_N are described by a structure of $q(X|R)$.

For many learning tasks, there is some priori knowledge about dependence structure among X_N, and thus we actually describe the dependence by some corresponding structure in the inner representation R. Several types of basic dependence have been discussed in Sect. 3.2 as a part of Ying structure design. This section provides further insights on several composite structures.

5.1 Lattice mode-switching factor analysis

5.1.1 Independent FA: Gaussian FA, NFA, and BFA

As shown in Fig. 7, each observation x comes from its de-noised counterpart $\hat{x}$ that is generated from latent factors, and each factor comes from a column of cells on a lattice, with each cell featured by both a real random variable $y_{j\ell}^r$ from $q(y_j^r|\theta_{j\ell})$ and a binary random variable $y_{j\ell}^z$ that takes either 1 to activate or 0 to switch off the corresponding cell. The variables of $y_{j\ell}^r$ are independent from each other across different cells. Each column sums up the outputs of activated cells to generate $\hat{x}$. The activating binary variables come from

$$q(y^z|\alpha^z) = \begin{cases} \frac{\prod_{j,\ell} {\alpha_{j\ell}^z}^{y_{j\ell}^z}}{\sum_{y_{j\ell}^z \in Y_s^z} \prod_{j,\ell} {\alpha_{j\ell}^z}^{y_{j\ell}^z}}, & \text{for } y_{j\ell}^z \in Y_s^z, \\ 0, & \text{otherwise,} \end{cases} \tag{76}$$

which consists of independent contributions from different cells and also a global constraint by the denominator that not only normalizes each individual contribution but also shapes a gating structure Y_s^z.

We start from the following typical case:

$$Y_s^z = \{y_{j\ell}^z : \sum_\ell y_{j\ell}^z = 1\}, \tag{77}$$

which gates the cells of each column as a stochastic

switch, resulting in

$$q(y^r) = \prod_j q(y_j^r), \text{with } y_j^r \in R \text{ from}$$
$$q(y_j^r) = \sum_\ell q(y_j^r|\theta_{j\ell})\alpha_{j\ell}^z,$$
$$\sum_\ell \alpha_{j\ell}^z = 1,\ 1 \geqslant \alpha_{j\ell}^z \geqslant 0. \tag{78}$$

One most widely considered case is

$$q(y_j^r|\theta_{j\ell}) = G(y_{j\ell}^r|\mu_{j\ell}, \lambda_{j\ell}). \tag{79}$$

Subject to an additive white noise e, the observation x is generated from the factors $y^r = \{y_j^r\}$ via a post-linear system $x = \eta(\hat{x}) + e$ as shown in Fig. 7, see Sect. 2 in Ref. [3] for a recent systematic introduction.

In the special case $\eta(\hat{x}) = \hat{x}$ with e from $G(e|0, \Psi)$ and $q(a_{ij}|y_{ij}^a) = q(a_{ij})$ (i.e., without gated by y_{ij}^a), we are lead to a general formulation for several types of independent factor analysis.

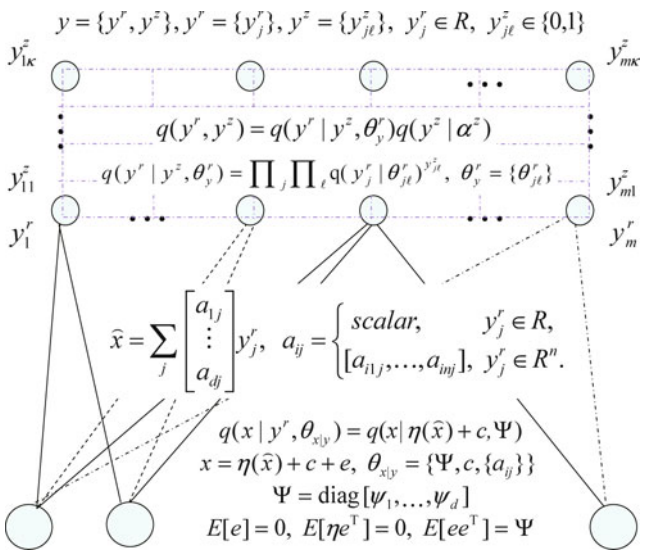

Fig. 7 Mode-switching lattice factor analysis

The simplest case $\kappa = 1$ has $q(y^r) = G(y^r|\mu, \Lambda)$ with $\Lambda = \text{diag}[\lambda_1, \lambda_2, \ldots, \lambda_m]$, which leads to FA-b by Eqs. (25) and (26), and to FA-a when $\mu = 0, \Lambda = I$.

A general formulation with $\kappa \geqslant 1$ has been previously studied under the name of non-Gaussian factor analysis (NFA). Details about NFA are referred to a paragraph between Eq. (82) and Eq. (83) in Ref. [40], and also to Sect. 5.2 in Ref. [47] and Sect. IV(C) in Ref. [4].

When $\kappa = 2$ and $\ell = 1, 2$, it degenerates to a BFA [47,65,77] at a special setting that $\mu_{j,1} = 1, \mu_{j,0} = 0$, and $\lambda_{j,1} = \lambda_{j,0} = 0$. With $\lambda_{j,1}, \lambda_{j,0}$ further becoming unknown parameters, we are lead to a noisy BFA, featured with that y_j^r taking binary values $0, 1$ becomes taking real vales from Gaussians centered at 0 and 1, respectively. Moreover, two Gaussians may be placed elsewhere with $\mu_{j,1}, \mu_{j,0}$ to be learned, which leads to a bipolar extension of FA-a and FA-b.

The two parts in $x = \hat{x} + e$ may be extended. First, a post-linear function $\eta(r)$ may be element-wisely added on $\hat{x} = Ay + c$, like those of the exponential family in the studies of generalized linear model [78], e.g., see Eqs. (21) and (22) in Ref. [3]. Second, the linear $\hat{x} = Ay + c$ may be extended into a quadratic function:

$$\hat{x} = [\hat{x}^{(1)}, \hat{x}^{(2)}, \ldots, \hat{x}^{(d)}]^{\mathrm{T}},$$
$$\hat{x}^{(i)} = c^{(i)} + \sum_j a_{ij} y_j^r + \sum_{j,\ell} b_{j\ell}^{(i)} y_j^r y_\ell^r. \tag{80}$$

5.1.2 Semi-blind learning and semi-blind FA

As suggested in Sect. 1 (see page 89) of Ref. [3], the term semi-supervised learning refers to efforts that put attention on a general scenario of knowing partially either or both of system and latent factors. One simplest case is that some elements in A are zero while others are unknown to be estimated, which leads to variants of networks component analysis (NCA) [20,21]. Moreover, with help of the following prior:

$$q(A, L) = \prod_{ij} q(a_{ij}|\ell_{ij}) q(\ell_{ij}),$$
$$q(\ell_{ij} = 1) = \beta_{ij},\ q(\ell_{ij} = 0) = 1 - \beta_{ij}, \tag{81}$$

a hard switching off $a_{ij} = 0$ is further relaxed to probabilistical switching between the mode $\beta_{ij} q(a_{ij}|\ell_{ij} = 1)$ and the mode $(1 - \beta_{ij}) q(a_{ij}|\ell_{ij} = 0)$.

Also, there are scenarios of knowing partially latent factors. One example is discrete feature based classification, that is, building a classifier $y_t \to x_t$ with x_t taking class labels and with a feature vector $y_t = [y_t^{(1)}, y_t^{(2)}, \ldots, y_t^{(m)}]^{\mathrm{T}}$ consisting of several discrete features. Such tasks are widely encountered in statistical data from social studies and gene analysis.

Traditionally, each discrete feature $y_t^{(j)}$ is treated as a real valued variable and discriminative boundaries are built in a real vector space R^m. One way is simply using $y_t^{(j)} = 1, 2, \ldots, \kappa$ as the values. However, such a uniformly spaced integer may not describe the class structure well. A better way is letting $y_t^{(j)}$ to take one of $r_{j1}, r_{j2}, \ldots, r_{j\kappa}$ real values. However, a difficulty is how to specify these real values.

Here, we reformulate this supervised learning into a simplified NFA learning given in Fig. 7. We encode x_t to take class labels by

$$[x_t^{(1)}, x_t^{(2)}, \ldots, x_t^{(d)}]^{\mathrm{T}}, \text{ with}$$
$$x_t^{(i)} = \begin{cases} 1, & x_t = i, \\ 0, & \text{otherwise}, \end{cases} \text{ and } \sum_i x_t^{(i)} = 1, \tag{82}$$

and such a vector comes from $q(x_t|y^r, \theta_{x|y}) = q(x_t|u, \theta_{x|y})|_{u=y^r}$ as follows:

$$q(x_t|u, \theta_{x|y}) = \prod_i \eta^{(i)}(\hat{x}_t)^{x_t^{(i)}},\ \hat{x}_t = Au + a_0,$$

$$\eta(\hat{x}_t) = [\eta^{(1)}(\hat{x}_t), \eta^{(2)}(\hat{x}_t), \ldots, \eta^{(d)}(\hat{x}_t)]^{\mathrm{T}},$$
$$\eta^{(i)}(\hat{x}_t) = \frac{1}{1 + \sum_{j \neq i} e^{\hat{x}_t^{(j)} - \hat{x}_t^{(i)}}}. \tag{83}$$

It follows from Eqs. (78) and (79) that a discrete feature $y_t^{(j)} = 1, 2, \ldots, \kappa$ corresponds to $y^z_{j\ell,t}$ taking 0 or 1, subject to $\sum_\ell y^z_{j\ell,t} = 1$. That is, we let

$$y^z_{j\ell,t} = \begin{cases} 1, & y_t^{(j)} = \ell, \\ 0, & \text{otherwise}, \end{cases} \tag{84}$$

while there are the following options for considering $r_{j1}, r_{j2}, \ldots, r_{j\kappa}$:

- Let $\mu_{j\ell} = \ell$, $\ell = 1, 2, \ldots, \kappa$, and every $\lambda_{j\ell}$ to be zero or a small positive number. This setting simulates a discrete feature to take $y_t^{(j)} = 1, 2, \ldots, \kappa$ but with probabilities $\alpha^z_{j\ell}$, $\ell = 1, 2, \ldots, \kappa$;
- Still set every $\lambda_{j\ell}$ to be zero or a small positive number, but let each $r_{j\ell} = \mu_{j\ell}$ to take a real value estimated via learning;
- Relax each $\mu_{j\ell}$ and $\lambda_{j\ell}$ to take real values estimated via learning, such that $r_{j\ell}$ stochastically comes from $G(y^r_{j\ell}|\mu_{j\ell}, \lambda_{j\ell})$.

5.1.3 Lattice Gaussian mixture and local FA

We consider another typical case as follows:

$$Y^z_s = \{y^z_{j\ell} : \sum_{j\ell} y^z_{j\ell} = 1\}, \tag{85}$$

which gates all the cells on the lattice as a stochastic switch, with each cell associated with a random vector $y^r_j \in R^n$ from a Gaussian distribution by Eq. (79) and accordingly $A = [a_{ij}]$ is extended into $A = [a_{i\ell j}]$. At the case $\eta(\hat{x}) = \hat{x} + c$, it follows from Eq. (85) that

$$\hat{x} = A_j y^r_j + c, \quad \text{with } y^r_j \text{ from } G(y^r_j|\mu_{j\ell}, \Lambda_{j\ell}) \text{ if } y^z_{j\ell} = 1, \tag{86}$$

and we observe that $x = \hat{x} + e$ comes from the following lattice Gaussian mixture:

$$q(x) = \sum_{j\ell} \alpha^z_{j\ell} G(x|c_{j\ell}, \Sigma_{j\ell}),$$
$$\sum_{j\ell} \alpha^z_{j\ell} = 1, \ 1 \geqslant \alpha^z_{j\ell} \geqslant 0, \tag{87}$$

which is actually a two-layer hierarchical Gaussian mixture (e.g., see Fig. 12 in Ref. [1]), and further degenerates to a standard Gaussian mixture when $\kappa = 1$. Some insights are obtained from the following special cases:

- The simplest one comes from the special case that

$$n = d, \ c = 0, \quad a_{i\ell j} = \begin{cases} 1, & \text{if } i = \ell, \\ 0, & \text{otherwise}, \end{cases} \tag{88}$$

at which we have

$$c_{j\ell} = \mu_{j\ell}, \ \Sigma_{j\ell} = \Psi + \Lambda_{j\ell}. \tag{89}$$

- A may be relaxed as follows

$$A_j = A, \quad \text{for all } j, \tag{90}$$

which leads to a lattice Gaussian mixture on a subspace spanned by the column vectors of A, that is, Eq. (89) becomes

$$c_{j\ell} = A\mu_{j\ell}, \ \Sigma_{j\ell} = \Psi + A\Lambda_{j\ell}A^{\mathrm{T}}. \tag{91}$$

- Equation (90) may be further relaxed by letting each A_j to be free individually, which leads to m subspaces with each spanned by the columns of its own A_j. The probability of $y^z_{j\ell} = 1$ is $\alpha^z_{j\ell}$, at which y^r_j comes from Eq. (86) as the input of

$$q(x|y^r, y^z, \theta_{x|y}) = G(x|A_j y^r_j, \Psi). \tag{92}$$

Therefore, Eq. (91) is written into

$$c_{j\ell} = A_j\mu_{j\ell}, \ \Sigma_{j\ell} = \Psi + A_j\Lambda_{j\ell}A_j^{\mathrm{T}}. \tag{93}$$

- Actually, this case together with Eq. (79) forms a lattice mixture of $m \times \kappa$ local FA models with each located at $A_j\mu_{j\ell}$ and spanned by the columns of A_j. When $\kappa = 1$, it degenerates to an LFA previously introduced in Eq. (28).
- Instead of locating on each local subspace, we may also consider that all the Gaussian centers locate on another subspace, that is, we may also consider $c \neq 0$ that

$$\mu_{j\ell} = 0, \quad c_{j\ell} = A\nu_{j\ell} + c, \tag{94}$$

with unknown parameters $A, \nu_{j\ell}, c$ all estimated during learning [79]. On one hand, it releases the constraint that each Gaussian center is bundled with one local FA model. On the other hand, it reduces the number of free parameters when the dimension of x is very high.

5.1.4 BYY harmony learning

Given $q(y^r, y^z)$ and $q(x_t|y^r, \theta_{x|y})$ in Fig. 7, it follows from Eq. (61) that

$$H_t(p||q, \theta) = \sum_{y^z \in D^*_{\eta^z}} \int p(y^r, y^z|x_t, \theta_p)\pi_t(y^r, y^z)\mathrm{d}y^r$$
$$\approx \sum_{y^z \in D^*_{\eta^z}} p_\gamma(y^z|x_t)\pi_t(y_t^{*r}, y^z)$$
$$-\frac{1}{2}\sum_{y^z \in D^*_{\eta^z}} p_\gamma(y^z|x_t)\mathrm{Tr}[\varepsilon_t\varepsilon_t^{\mathrm{T}}\Pi^q_{y|x,\theta} + \Gamma^{y|x}_\Pi],$$
$$p(y^r, y^z|x_t, \theta_p) = p_\gamma(y^z|x_t)p(y^r|y^z, x_t),$$
$$\pi_t(y^r, y^z) = \ln[q(x_t|y^r, \theta_{x|y})q(y^r|y^z, \theta^r_y)q(y^z|\alpha^z)q(\theta)],$$
$$[y_t^{*r}, y_t^{*z}] = \arg\max_{y^r, y^z} \pi_t(y^r, y^z),$$

$$\varepsilon_t = y_t^{*r} - \eta_y(x_t),$$
$$\Gamma_{\Pi}^{y|x} = \Gamma_{y|x,\theta}^{p} \Pi_{y|x,\theta}^{q}, \tag{95}$$

where $\Gamma_{y|x,\theta}^{p}$, $\Pi_{y|x,\theta}^{q}$, and $\eta_y(x_t)$ are given in Table 2 of Type S, covering both only x_t available and a given input-output pair $x_t \to y_t$.

Taking a same role as j_t^* in Eq. (14), we have

$$\eta^z = \begin{cases} y_t^{*r}, & \text{with only } x_t \text{ available,} \\ y_t, & \text{for each supervised pair } x_t, y_t. \end{cases} \tag{96}$$

Accordingly, $D_{\eta^z}^*$ is the apex-zone centered at η^z with y^z taking values in a neighborhood of η^z. One extreme end is that $D_{\eta^z}^*$ merely consists of η^z, while the other extreme is that $D_{\eta^z}^*$ is the entire domain of y^z.

Similar to $q_\gamma(\ell|x_t,\theta)$ in Eq. (14), we have

$$p_\gamma(y^z|x_t) = \frac{\gamma\delta_{y^z,y_t^{*z}} + q(y^z|\alpha^z)q(x_t|y^z,\theta_{x|y},\theta_y^r)}{\gamma + \sum_{y^z} q(y^z|\alpha^z)q(x_t|y^z,\theta_{x|y},\theta_y^r)},$$
$$q(x_t|y^z,\theta_{x|y},\theta_y^r) = \int q(x_t|y^r,\theta_{x|y})q(y^r|y^z,\theta_y^r)\mathrm{d}y^r \approx |\Pi^r(y^z)|^{-0.5}[q(x_t|y^r,\theta_{x|y})q(y^r|y^z,\theta_y^r)]_{y^r=y_t^{*r}(y^z)},$$
$$y_t^{*r}(y^z) = \arg\max_{y^r}[q(x_t|y^r,\theta_{x|y})q(y^r|y^z,\theta_y^r)], \tag{97}$$

where $\gamma\delta_{y^z,y_t^{*z}}$ is a generalized format of $\gamma\delta_{\ell,j_t^*}$ in Eq. (14). Again, the precision parameter γ in $p_\gamma(y^z|x_t)$ controls the teaching degree in semi-supervised learning, with unsupervised learning at one extreme $\gamma = 0$ that the teaching label becomes completely useless and with supervised learning at the other extreme $\gamma = \infty$ that the teaching label is precisely correct. This $p_\gamma(y^z|x_t)$ simply provides a common formulation that facilitates to train a traditional supervised learning task via implementing semi-supervised learning.

Furthermore, $q(\theta)$ is an appropriate prior and it follows from the least redundancy principle that we assume the independence of parameters in each component as follows:

$$q(\theta) = q(\theta_{x|y})q(\theta_y^r)q(\alpha^z)q(\rho)q(\gamma),$$
$$q(\theta_y^r) = \prod_{j\ell} q(\theta_{j\ell}) = \prod_{j\ell}[q(\mu_{j\ell}^r)q(\lambda_{j\ell})],$$
$$q(\theta_{x|y}) = q(\Psi)q(\{a_{ij}\},c) = q(c)\prod_{ij} q(a_{ij})\prod_i q(\psi_i),$$
$$q(\alpha^z) = \prod_j q(\alpha_{j\ell}^z, \ell = 1,2,\ldots,\kappa),$$
$$q(\rho) = \prod_j q(\rho_j), \tag{98}$$

where we have respectively

Inverse Gamma : $q(\lambda_{j\ell}), q(\rho_j), q(\psi_i)$,
Gamma : $q(\gamma)$,
Dirichlet : $q(\alpha_{j\ell}^z, \ell = 1,2,\ldots,\kappa)$,
Gaussian : $q(c), q(a_{ij})$, with zero mean, e.g., $N(0,1)$,
Laplacian : $q(a_{ij})$ for sparse learning [80,81].

From Eqs. (61) and (95), we can get a Ying-Yang alternation algorithm for learning implementation. Some insights may be obtained from the following two examples. First, we start from the special case of Eq. (87) at $\kappa = 1$, i.e., a standard Gaussian mixture. In this case, it follows from Eqs. (61) and (95) that we have

$$H(p\|q,k,\theta,\Xi) = \sum\nolimits_t \sum\nolimits_{j\in J_t^\kappa} q_\gamma(j|x_t,\theta)\pi_t(\theta_j) + \ln[q(\gamma)\prod_j q(\theta_j)], \tag{99}$$

where $q_\gamma(j|x_t,\theta)$, $\pi_t(\theta_j)$, J_t^κ are same as the ones in Eq. (14). Considering the gradient $\nabla_\varphi H(p\|q,k,\theta,\Xi)$, we have

$$\nabla_\varphi H(p\|q,k,\theta,\Xi) = \sum\nolimits_t \sum\nolimits_{j\in J_t} p_{j,t}\nabla_\varphi \pi_t(\theta_j) + \nabla_\varphi \ln[q(\gamma)\prod_j q(\theta_j)], \tag{100}$$

from which we obtain $p_{\ell,t}$ in Eqs. (10) and (14). Learning is implemented by iteratively getting this $p_{\ell,t}$ and putting it to update θ^* by Eq. (3). Also, we may learn an appropriate γ with help of a prior $q(\gamma)$ of Gamma distribution.

Next, we observe another example of putting $q(x|y^r,\theta_{x|y}) = G(x|Ay^r + c, \Psi)$ into Eq. (86), which leads to the previous studies under NFA (e.g., see Sect. IV(C) in Ref. [4]). One key problem is to solve $\max_{y^r,y^z} \pi_t(y^r,y^z)$. E.g., getting $y_t^{*r}(y^z)$ and solving

$$y_t^{*z} = \arg\max_{y^z}[q(y_t^{*r}(y^z)|y^z,\theta_y^r)q(y^z|\alpha^z)]. \tag{101}$$

While $y_t^{*r}(y^z)$ is analytically obtained as follows:

$$y_t^{*r}(y^z) = [A^{\mathrm{T}}\Psi^{-1}A + \Lambda_{y^z}^{-1}]^{-1}[A^{\mathrm{T}}\Psi^{-1}(x-c) + \Lambda_{y^z}^{-1}\mu_{y^z}]. \tag{102}$$

Beyond the previous studies, we are also provided with the following features:

- the regularization roles of $\varepsilon_t^{\mathrm{T}}\Pi_{y|x,\theta}^q\varepsilon_t$ and $\mathrm{Tr}[\Gamma_{\Pi}^{y|x}]$ in consideration;
- a prior $q(\theta)$ in consideration;
- an extension of the linear $\hat{x} = Ay$ into the quadratic function in Fig. 7.

5.2 Piecewise stationary temporal structure

5.2.1 TFA and extensions

Equations (95) and (61) come from Eqs. (53) and (57) for the samples of $X_N = \{x_t\}$ that are independent and identically distributed (i.i.d.). For samples with temporal dependence, as discussed around Eq. (30), it is preferred to use $q(Y)$ with temporal structure for this purpose though we may use either or both of $q(Y)$ and

$q(X|R)$ to capture this type of dependence. Typically, $q(Y)$ is considered to be Markovian (e.g., the first order Markovian) while $q(X|R)$ is still instantaneous as in Eq. (61). That is, we consider

$$\begin{aligned} q(Y|\theta) &= q(y_0)\prod_{t\geqslant 1} q(y_t|y_{t-1},\theta_y),\\ q(X_N|Y,\theta) &= \prod_{t\geqslant 0} q(x_t|y_t,\theta_{x|y}),\\ H(p||q,\theta) &= H_0(p||q,\theta)+\sum_{t\geqslant 1} H_t(p||q,\theta),\\ H_t(p||q,\theta) &= \int p(y_t,y_{t-1}|x_t)\pi_t(y_t,y_{t-1},\theta)\mathrm{d}y_t\mathrm{d}y_{t-1},\\ \pi_t(y_t,y_{t-1},\theta) &= \ln[q(x_t|y_t,\theta_{x|y})q(y_t|y_{t-1},\theta_y)q(\theta)]. \end{aligned} \tag{103}$$

For a sample set with a large N, we may ignore the term $H_0(p||q,\theta)=\int p(y_0|x_0)\ln[q(x_0|y_0,\theta_{x|y})q(y_0|\theta_y)q(\theta)]\mathrm{d}y_0$.

As suggested at the end of Sect. 5.1 in Ref. [1], the integral over $\mathrm{d}y_t\mathrm{d}y_{t-1}$ can be solved

- analytically when $q(x_t|y_t,\theta_{x|y})$ and $q(y_t|y_{t-1},\theta_y)$ are both Gaussian, and thus it follows from Eq. (60) in Ref. [1] that $p(y_t,y_{t-1}|x_t)=q(y_t,y_{t-1}|x_t)$ is also Gaussian,
- or in help of apex approximation by Eq. (55), e.g., by the schematic algorithm as shown in Fig. 13 in Ref. [1].

One typical example of $q(x_t|y_t,\theta_{x|y})$ is still the instantaneous linear relation by Eq. (25), while $q(y_t|y_{t-1},\theta_y)$ is given by Eq. (30), as shown in Fig. 8. It has been referred under the name of TFA or independent state space (ISS) models. Originated from Refs. [63,66], TFA studies [5,40,47,61,71,74] extend the classic FA model by Eq. (25) through taking temporal dependence into consideration by Eq. (30) with both B,Λ being diagonal in order to keep the cross-dimensional independence of y_t. Taking the observation noise in consideration by Eq. (25), TFA also differs from those efforts for implementing temporal independent component analysis (TICA), e.g., joint diagonalization, context sensitive ICA, and temporal BYY harmony learning based temporal ICA, etc. Details are referred to Sect. 6.1 in Ref. [47].

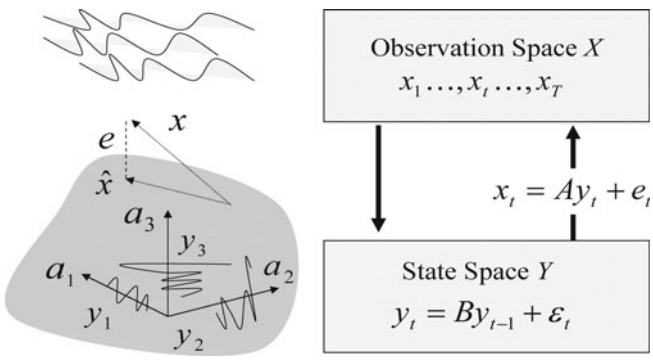

Fig. 8 Temporal factor analysis

Without constraining B,Λ to be diagonal, Eqs. (25) and (30) will become a general state space model (SSM) or a linear dynamical system (LDS) widely studied in the literature of control theory and signal processing [82]. In a period that is more or less the same period as the studies on TFA [61,63,66,71,74], there was a renewed interest on the general SSM or LDS, featured by using the EM algorithm for parameter estimation under the maximum likelihood principle [83,84]. This EM algorithm was originally derived by Shumway and Stoffer [85,86], and re-introduced in the early 1990's [86,87]. Being obviously different from those studies of control system theory, how to make a model become identifiable and stable was unfortunately out of consideration in these renewed efforts, though parameter estimation was occasionally mentioned under the term system identification [83,84].

On the contrary, TFA studies [5,40,47,61] aim at a model with a guaranteed stability and an improvement on identifiability. Favorably, it has been shown in Sects. III and IV of Ref. [61] that Eqs. (25) and (30) indeed improve the identifiability of the FA model by Eq. (25) because the notorious rotation indeterminacy of the classic FA has been removed due to Eq. (30) with a diagonal matrix $B\neq 0$. In Ref. [72], not only the stability of TFA is ensured with each diagonal element of B satisfying $|b_i|<1$, but also an identifiable family of TFA structures has been investigated.

Applied to radar automatic target recognition based on high-resolution range profile (HRRP) as shown in Fig. 9, it has been empirically shown (see Table 4 in Ref. [88]) that the recognition performance of the general SSM or LDS is actually even inferior to that of the classic FA due to many extra free parameters, which makes identifiability become even worse. On the contrary, TFA obtains better recognition and rejection performances than the classic FA because of considering temporal correlation.

Also, there is another difficult task of selecting an appropriate hidden dimension m of y_t, the studies in Refs. [83–87] use the EM algorithm to implement the maximum likelihood learning, for which selecting an appropriate m needs a computational intensive two-stage implementation with help of a model selection criterion such as AIC or BIC. In contrast, the TFA studies [5,40,47,61] perform the BYY harmony learning, by which model selection is made either automatically during learning or still in a two-stage implementation but with an improved performance by a BYY harmony selection criterion. On HRRP radar target recognition, it has also been empirically shown in Ref. [88] that the BYY harmony learning based FA and TFA further outperform the two-phase learning based TFA in both estimation accuracy and computational efficiency.

Moreover, we may consider the following TFA extensions with $q(x_t|y_t,\theta_{x|y})$ and $q(y_t|y_{t-1},\theta_y)$ in other choices:

1) Temporal BFA: When $y_{t,\ell}$ is a binary vector with each element $y_{t,\ell}^{(j)}$ taking either 0 or 1. Instead of $y_t = By_{t-1} + \varepsilon_t$, we consider

$$q(y_t|y_{t-1},\theta_y) = \prod_j s^{y_t^{(j)}}(\hat{y}_t^{(j)})[1-s(\hat{y}_t^{(j)})]^{1-y_t^{(j)}},$$
$$\hat{y}_t^{(j)} = \sum_{\tau=1}^{\kappa} b_{j,\tau} y_{t-\tau}^{(j)} + b_{0,\tau}, \quad (104)$$

where $0 < s(r) < 1$ is a sigmoid function, see the algorithm given in Fig. 8 of Ref. [1].

2) Temporal NFA: When $y_{t,\ell}$ comes from a mixture of scalar Gaussians, i.e., by Eqs. (78) and (79), we get a temporal extension of NFA. Readers are further referred to Refs. [4,5].

3) Efforts may also be made on extensions via different choices of $q(x_t|y_t,\theta_{x|y})$, e.g., the linear $\hat{x} = Ay + c$ may be extended into considering a quadratic function given by Eq. (80) as shown in Fig. 7.

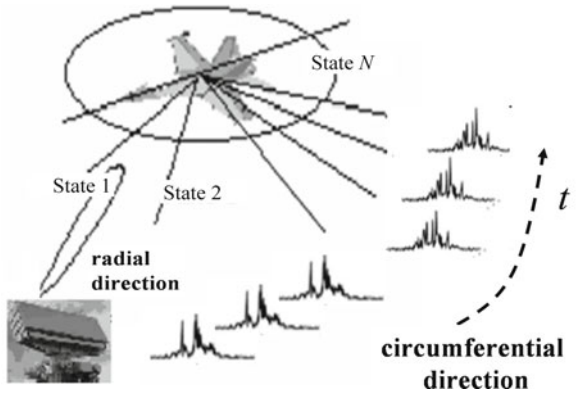

Fig. 9 Radar automatic target recognition

5.2.2 HMM gated TFA and extensions

The TFA or ISS modeling shown in Fig. 8 targets at stationary temporal dependence among samples. In many real applications, we encounter temporal dependence that is stationary within a certain length of segment but switches to different statistical properties across different segments. Taking the task in Fig. 9 as an example, radar HRRP returns are featured by three types of dependence. One is spatial dependence along radial direction, i.e., inter-dimensional dependence of x_t. As radar rotates, HRRP returns are divided into a number of sections, called aspect frames. There is a stationary temporal dependence among samples that come from a same aspect frame. However, statistical property may alter or even suddenly change across different aspect frames. We need an intrinsic framework for modeling a temporal sequence $x_1,\ldots,x_t,\ldots,x_N$ that is divided into many segments such that inter-dimensional dependence of x_t, stationary temporal dependence among samples within an aspect frame, and long range across-frame temporal dependence are all appropriately modeled.

For the above modeling framework, we consider HMM gated TFA modeling. As shown in Fig. 9, an HMM model is considered for modeling long range across-frame temporal dependence, with each hidden state associated with a TFA model for both stationary temporal dependence among samples within a segment and inter-dimensional dependence of x_t. This HMM gated TFA modeling probabilistically covers different segments without an explicit segmentation.

The hidden states are connected in certain structure, e.g., a line tri-phone structure shown in Fig. 10(a) with one or more Gaussians under each state, as widely used in speech recognition [89]. Also, such a structure may be used as shown in Fig. 10(b) for modeling the HRRP data in Fig. 9. This structure uses a self-circle per state to probabilistically describe a random length per segment, and uses a jump from one state to the next one to ensure the unidirectional nature.

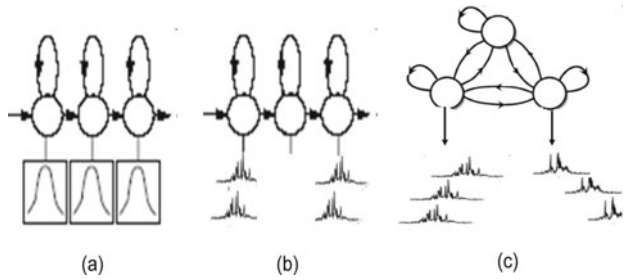

Fig. 10 Hidden states: Line versus cyclic structure

Alternatively, we may use a cyclic structure shown in Fig. 10(c) to model the circumferential feature of the HRRP data. All the unknown parameters in the HMM and each TFA model, as well as the number of hidden states and the state space dimension of each TFA model, are all learned from a given sequence $x_1,\ldots,x_t,\ldots,x_N$ by the BYY harmony learning.

The implementation of the BYY harmony learning on the above HMM gated TFA modeling is a special case of a general formulation given by Eqs. (57) and (58) in Ref. [1]. Ignoring the prior $q(\theta)$ and shutting off the data smoothing by letting $h = 0$, this formulation is simplified into $H(p||q,\theta) = \sum_t H_t(p||q,\theta)$ with

$$H_t(p||q,\theta) = H_t^{\mathrm{HMM}}(p||q,\theta) + \sum_{\ell_t} p(\ell_t|x_t,\theta) H_t^{\mathrm{ltfa}}(p||q,\ell_t,\theta),$$
$$H_t^{\mathrm{ltfa}}(p||q,\ell_t,\theta) = \int p(y_t,y_{t-1}|\ell_t,x_t,\theta)\ln q(x_t,y_t|y_{t-1},\ell_t,\theta)\mathrm{d}y_{t-1}\mathrm{d}y_t,$$
$$H_t^{\mathrm{HMM}}(p||q,\theta) = \sum_{\ell_t,\ell_{t-1}} p(\ell_t,\ell_{t-1}|x_t,\theta)\ln q(\ell_t,\ell_{t-1}|Q),$$
$$q(x_t,y_t|y_{t-1},\ell_t,\theta) = q(x_t|y_t,\ell_t,\theta_{x|y,\ell_t}) q(y_t|y_{t-1},\ell_t,\theta_{y,\ell_t}). \quad (105)$$

Maximizing $H(p||q,\theta)$ is made via $\nabla_\varphi H_t(p||q,\theta) = \nabla_\varphi H_t^{\rm HMM}(p||q,\theta) + \sum_{\ell_t} H_t^{\rm ltfa}(p||q,\ell_t,\theta)\nabla_\varphi p(\ell_t|x_t,\theta) + \sum_{\ell_t} p(\ell_t|x_t,\theta)\nabla_\varphi H_t^{\rm ltfa}(p||q,\ell_t,\theta)$, where $\nabla_\varphi p(\ell_t|x_t,\theta) = \sum_{\ell_{t-1}} \nabla_\varphi p(\ell_t,\ell_{t-1}|x_t)$ and $\nabla_\varphi H_t^{\rm HMM}(p||q,\theta)$ comes from $\nabla_\varphi \ln q(\ell_t|\ell_{t-1},Q)$ and $\nabla_\varphi p(\ell_t,\ell_{t-1}|x_t,\theta)$, in a way similar to handling $\nabla_\varphi \ln \alpha_{j|\ell}$ and $\nabla_\varphi q(l_t,l_{t-1}|\theta)$ by the Box ② and Box ③ in Fig. 14(b) of Ref. [1] via $p(l_t,l_{t-1}|\theta) = q(l_t,l_{t-1}|\theta)$ in choice (c). Specifically, $\nabla_\varphi H_t^{\rm ltfa}(p||q,\ell_t,\theta)$ for each ℓ_t is handled in the same way as $\nabla_\varphi H_t(p||q,\theta)$ in Eq. (103).

Moreover, different structures of $q(x_t|y_t,\ell_t,\theta_{x|y,\ell_t})$ and $q(y_t|y_{t-1},\ell_t,\theta_{y,\ell_t})$ may lead to other types of the HMM gated temporal modeling. One typical example is $q(x_t|y_t,\ell_t,\theta_{x|y,\ell_t})$ by Eq. (28) and $q(y_t|y_{t-1},\ell_t,\theta_{y,\ell_t})$ by

$$\begin{aligned} &y_{t,\ell} = B_\ell y_{t-1,\ell} + \varepsilon_{t,\ell},\ \ Ey_{t-1,\ell}\varepsilon_{t,\ell}^{\rm T} = 0,\\ &\varepsilon_{t,\ell} \sim G(\varepsilon_{t,\ell}|0,\Lambda_\ell),\\ &B_\ell = {\rm diag}[b_{\ell,1},b_{\ell,2},\ldots,b_{\ell,m}]. \end{aligned} \tag{106}$$

Putting them into Eq. (105), we make the BYY harmony learning on the HMM gated TFA modeling, where $\int[\cdot]{\rm d}y_{t-1}{\rm d}y_t$ becomes analytically solvable.

Furthermore, efforts are also needed on extensions to the HMM gated temporal BFA modeling and the HMM gated temporal NFA modeling as well as other choices as discussed around Eq. (104).

5.3 Hierarchical and graphical structure

5.3.1 Hierarchical structures

Also, we may consider an HMM gated mixture of nonstationary segments, while each nonstationary segment is itself modeled by a HMM gated mixture of stationary segments, which leads to a tree or hierarchical structure. Even for a set of i.i.d. samples without temporal dependence, a hierarchical structure is still helpful for effectively describing a complicated distributions.

As illustrated at the center in Fig. 11, we consider a two-level hierarchical mixture of Gaussians. For each sample x_t, we get a teaching label to assign x_t to a Gaussian mixture, which belongs to supervised learning. However, it becomes an unsupervised learning task as we further assign x_t to one particular Gaussian component in this mixture. This problem can be regarded as a semi-supervised learning task in a sense that each sample has two teaching labels. One is known while the other is unknown to be determined. Alternatively, it may be regarded as an example of unsupervised learning in a sense that each x_t missed its information from which Gaussian component.

The two-level hierarchical mixture of Gaussians in Fig. 11 is equivalent to the lattice Gaussian mixture by Eq.

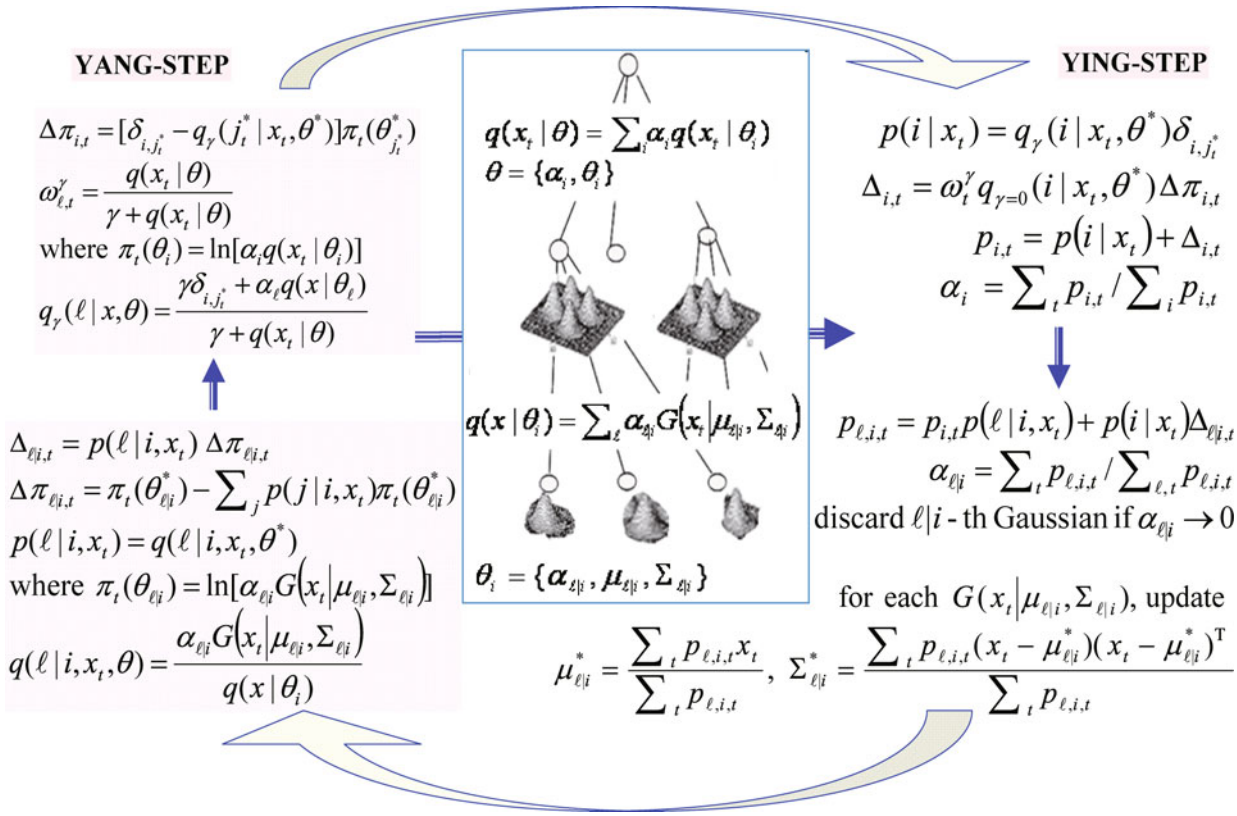

Fig. 11 A two-level hierarchical mixture of Gaussians and its Ying-Yang alternation learning algorithm

(87), simply with j indexing the top layer while ℓ indexing the bottom layer, as well as $\alpha_{j\ell}^z$ in place of $\alpha_j\alpha_{\ell|j}$ and $G(x|\mu_{j\ell},\Sigma_{j\ell})$ in place of $G(x|\mu_{\ell|j},\Sigma_{\ell|j})$.

Learning can be made by a Ying-Yang alternating algorithm as illustrated in Fig. 11 that implements hierarchically, with help of the chain rule for derivatives. The details are referred to Sect. 5.1 in Ref. [1] and especially its equation (55). As shown in Fig. 12 of Ref. [1], the learning algorithm is implemented by a hierarchical harmony flow from bottom-up by the Yang-step and top-down by the Ying-step. Precisely, learning a three-level hierarchical Gaussian mixture of is illustrated in Fig. 12 of Ref. [1].

Here, we simplify it to suit for a two-level hierarchy in Fig. 11. Still, updating $\alpha_{\ell|j},\mu_{\ell|j},\Sigma_{\ell|j}$ is similar to Eq. (3) except the indexes in a different notation. The key point is updating $p_{\ell,j,t}$ by the Ying-step in two levels. The top level makes supervised learning, while the bottom level makes unsupervised learning.

Moreover, we can combine the two-level hierarchical Gaussian mixture in Fig. 11 with the HMM gated TFA modeling introduced in the previous subsection, from which we get the following two temporal-hierarchical structures for further investigations:

1) Gaussian mixture based HMM and local FA mixture based HMM: Modifying the HMM gated TFA modeling by Eq. (105) with each local TFA modeling $q(x_t|y_t,\ell_t,\theta_{x|y,\ell_t})q(y_t|y_{t-1},\ell_t,\theta_{y,\ell_t})$ replaced by a two-level hierarchical mixture of Gaussians in Fig. 11 (i.e., under each hidden state there is a two-level hierarchical Gaussian mixture), we are lead to a Gaussian mixture based HMM model, as widely used in speech recognition literature [89]. Moreover, we may replace each Gaussian at the bottom level by a local FA model given by Eqs. (28) and (29), which leads to a further extension called local FA mixture based HMM. Readers are referred to Sect. 5.3 and Fig. 14 of Ref. [1] for an introduction and the corresponding BYY harmony learning algorithm.

2) HMM gated TFA models: We may further let the above local FA model replaced by a local TFA model with $q(x_t|y_t,\ell_t,\theta_{x|y,\ell_t})$ by Eq. (28) and $q(y_t|y_{t-1},\ell_t,\theta_{y,\ell_t})$ by Eq. (106) for describing each stationary but non-Gaussian segment. Alternatively, we may also let each local FA modeling to be replaced by another HMM gated TFA modeling for describing non-stationary temporal dependence in a hierarchical way.

5.3.2 Graphical structures

Beyond tree or hierarchical structures, another type of dependence is described by graphical structure. One typical example considers local topology described by a nearest neighbor graph, which has attracted lots of attentions in the past decade under the name of manifold learning. Given a sample data set $X_N=[x_1,x_2,\ldots,x_N]$, the key point is to get a graph Laplacian matrix L from a nearest neighbor graph G with each node denoting one column of X_N. Then, we define $q(Y)$ based on L. One example is given by Eq. (66) in Ref. [3].

As introduced in Sect. 2.2 and Fig. 5 of Ref. [1], updating $q(Y)$ takes an important role on automatic model selection by the BYY harmony learning.

In Ref. [3], when L is given, $q(Y)$ by Eq. (66) has no free parameter to be adjusted, such that the contribution by $q(Y)$ to automatic model selection is lost. This situation can actually be regarded as an extended counterpart of FA-a by Eqs. (25) and (27). Following the discussions made after Eq. (27), we are further motivated to consider the extended counterpart of FA-b by Eqs. (25) and (26). Considering a diagonal matrix Λ to take a role similar to that in $G(y_t|0,\Lambda)$, we propose to replace Eq. (66) in Ref. [3] with the following one:

$$q(Y)=\frac{1}{Z(L,\Lambda)}\exp\{-\frac{1}{2}\mathrm{Tr}[YLY^{\mathrm{T}}\Lambda^{-1}]\},$$
$$Z(L,\Lambda)=\int\exp\{-\frac{1}{2}\mathrm{Tr}[YLY^{\mathrm{T}}\Lambda^{-1}]\}\mathrm{d}Y. \quad (107)$$

Making manifold learning with Eq. (66) in Ref. [3] is accordingly replaced by Eq. (107), with help of a matrix L that is either a graph Laplacian matrix from X_N or a matrix that describes certain connectivity among columns of Y.

6 Gene analysis applications

6.1 Genome-wide association study

One recent popular topic in gene analysis is searching for genetic factors that influence common complex traits and the characterization of the effects of those factors. Most efforts are made on how SNPs influence complex traits (especially diseases) under the name of GWA study [90–93], where SNP is a shorthand of single nucleotide polymorphism. There are three genotypes for each SNP, and each SNP is represented by a discrete variable $y^{(j)}$ that takes one of three labels or discrete variables $y^{(j)}$, $j=1,2,\ldots,m$, with $m=3$, while whether a trait or disease manifests can be expressed by a binary variable $x=0,1$.

Typically, a logistic regression is used for modeling the probability of having a disease (i.e., $x=1$) as follows:

$$p(x=1)=\eta(\hat{x}),\quad \hat{x}=\sum_j a_j y^{(j)}+a_0,$$
$$\eta(u)=\frac{1}{1+e^{-u}}. \quad (108)$$

The parameters $a_0,a_1,\ldots,a_m$ are estimated by the ML learning, based on paired samples of $\{y_t,x_t\}$. Beyond

the ML learning, we may simply treat this problem by a simplified NFA model by Eq. (83) at the special case $d = 1$, featured with each discrete variable $y_t^{(j)}$ replaced by a real value $\mu_{j\ell}$ or $y_{j\ell}^r$.

Moreover, we may modify Eq. (83) as follows:

$$q(x_t|u,\theta_{x|y}) = \prod_i \eta(\hat{x}_t^{(i)})^{x_t^{(i)}}[1-\eta(\hat{x}_t^{(i)})]^{1-x_t^{(i)}},$$
$$\hat{x}_t = Au + a_0, \tag{109}$$

and then extend this simplified NFA learning for analyzing the relations between a set of SNPs and multiple complex traits, from which we observe how some SNPs simultaneously affect more than one traits or diseases.

Also, efforts have been extensively made on considering whether a complex trait is influenced by interactions between two SNPs. That is, a linear regression $\hat{x} = \sum_j a_j y^{(j)} + a_0$ is extended to $\hat{x} = \sum_j a_j y^{(j)} + \sum_{ij} b_{ij} y^{(i)} y^{(j)} + a_0$ with the second order terms. Statistical test is made on checking whether $b_{ij} = 0$ to verify whether the interaction between the corresponding two SNPs has influenced the trait.

According to the least complexity principle, each quadratic term is considered only when we have to. Efforts are needed to seek a learning mechanism that extra parameters are pushed towards zeros such that the coefficients of the quadratic terms are pushed more strongly than the coefficients of the linear terms are pushed. One way is considering appropriate priors $q(a_j)$ and $q(b_{ij})$.

Again, the above simplified NFA learning can be applied, with help of $\hat{x}$ given by Eq. (80). Together with Eq. (109), we may further observe how such interactions between two SNPs simultaneously affect more than one traits or diseases.

Moreover, checking $b_{ij} = 0$ for every SNP pair is computationally extensive. Alternatively, the above simplified NFA learning can be implemented via sparse learning on b_{ij}, with $\prod_{ij} q(a_{ij})$ replaced by $\prod_{ij} q(a_{ij}) \prod_{ij\ell} q(b_{j\ell}^{(i)})$ such that the automatic model selection nature pushes most of parameters $\{b_{j\ell}^{(i)}\}$ towards zero.

As illustrated in Fig. 12, we consider samples of two SNPs represented by discrete variable taking three labels. The logistic regression by Eq. (108) works on samples of linear separable as illustrated in Fig. 12(a1) but fails on samples of not linear separable as illustrated in Fig. 12(a2). Samples illustrated in Figs. 12(b1) and 12(c1) are also not linear separable, on which the logistic regression by Eq. (108) fails too. However, the above semi-blind NFA learning will estimate $\mu_{j\ell}$ to move samples slightly away from knots such that samples become separable as illustrated in Figs. 12(b2) and 12(c2).

Not only this semi-blind NFA learning improves the performance of the logistic regression by Eq. (108), but also finding and verifying such an improvement provides an alternative way that indicates whether the interaction between two SNPs has influenced the trait.

In implementation, we may first get the logistic regression by Eq. (108), and then make the semi-blind NFA learning by setting the resulted coefficients of $\{a_j\}$ as an initialization of A. Moreover, the updating direction of $\mu_{j\ell}$ can be modified towards its corresponding class, e.g., towards its class center along a projection of $\nabla_{\mu_{j\ell}} H(p\|q,k,\theta,\Xi)$ on the sample. In this way, this semi-blind NFA learning actually identify a subset of quadratic separable ability.

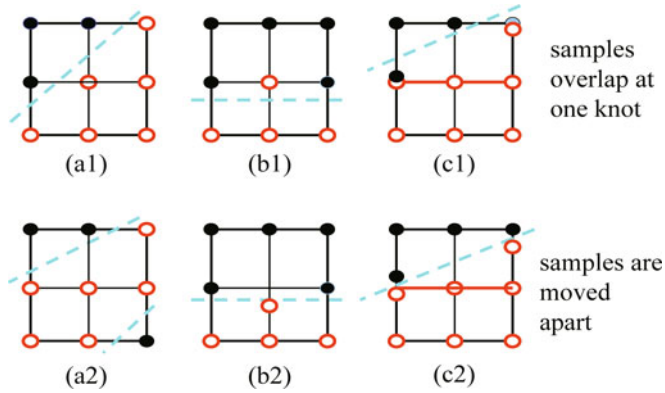

Fig. 12 Linear separable and extensions

Another method for GWA study is using a three-layer network for learning a mapping $\xi_t \to y_t \to x_t$ from paired samples $\{\xi_t, x_t\}$, again with help of the BYY harmony learning. A latest outline is referred to Sect. 4.4 and especially Eq. (51) in Ref. [1].

Such a three-layer network is learned as a special type of binary FA, with $q(y^z|\alpha^z)$ by Eq. (76) and $q(x_t|y^z,\theta_{x|y}) = q(x_t|u,\theta_{x|y})|_{u=y^z}$ by Eq. (109). For $\ell = 1,2$, we have

$$y_j^z = y_{j1}^z,\ y_{j2}^z = 1 - y_j^z,\ j = 1,2,\ldots,m,$$
$$\alpha_j^z = \alpha_{j1}^z,\ \alpha_{j2}^z = 1 - \alpha_j^z, j = 1,2,\ldots,m. \tag{110}$$

Simplified from Eq. (95), we further have

$$H_t(p||q,\theta) \approx \sum_{y^z \in D^*_{y^z}} p(y^z|\xi_t)\pi_t(y^z),$$
$$\pi_t(y^z) = \ln[q(x_t|y^z,\theta_{x|y})q(y^z|\alpha^z)q(\theta)],$$
$$q(\theta) = q(\theta_{x|y})q(\alpha^z)q(W,w),$$
$$y_t^{*z} = \arg\max_{y^z} \pi_t(y^z),$$
$$q(y^z|\alpha^z) = \prod_j [\alpha_j^z]^{y_j^z}[1-\alpha_j^z]^{1-y_j^z},$$
$$p(y^z|\xi_t) = \prod_j \eta(r_t^{(j)})^{y_j^z}[1-\eta(r_t^{(j)})]^{1-y_j^z},$$
$$r_t = W\xi_t + w,\ \eta(r) = \frac{1}{1+e^{-r}},$$
$$\eta(\xi_t) = [\eta(r_t^{(1)}), \eta(r_t^{(2)}), \ldots, \eta(r_t^{(m)})]^{\mathrm{T}}, \tag{111}$$

where $D^*_{y^z}$ is the apex-zone centered around y_t^{*z} by one or a few bits.

Alternatively, from Eq. (55) we approximately have

$$\begin{aligned} H_t(p||q,\theta) &\approx \pi_t(y_t^{*z}) - \frac{1}{2}\mathrm{Tr}[\varepsilon_t\varepsilon_t^{\mathrm{T}}\Pi^q_{y|x,\theta} + \Gamma^p_{y|x,\theta}\Pi^q_{y|x,\theta}], \\ \varepsilon_t &= y_t^{*z} - \eta_y(\xi_t), \\ \eta(\xi_t) &= [\eta(r_t^{(1)}), \eta(r_t^{(2)}), \ldots, \eta(r_t^{(m)})]^{\mathrm{T}}, \\ \Pi^q_{y|x,\theta} &= -\frac{\partial^2\pi_t(y^z)}{\partial y^z \partial y^{z\,\mathrm{T}}}, \\ \Gamma^p_{y|x,\theta} &= \mathrm{diag}[\eta(r_t^{(1)}), \eta(r_t^{(2)}), \ldots, \eta(r_t^{(m)})] \\ &\quad -\eta(\xi_t)\eta(\xi_t)^{\mathrm{T}}. \end{aligned} \tag{112}$$

Again, an algorithm can be developed to maximize $H(p||q,\theta) = \sum_t H_t(p||q,\theta)$ for implementing the BYY harmony learning. After learning, the mapping $\xi_t \to y_t \to x_t$ can be implemented via

$$p(x_t|\xi_t) = \sum_{y^z} q(x_t|y^z, \theta_{x|y}) q(y^z|\alpha^z) p(y^z|\xi_t), \tag{113}$$

or approximately via

$$\begin{aligned} &f(\xi) = \eta(u),\ \eta(u) = [\eta(u^{(1)}), \eta(u^{(2)}), \ldots, \eta(u^{(m)})], \\ &u = A\ \mathrm{diag}[\alpha_1^z, \alpha_2^z, \ldots, \alpha_m^z]\ \eta(W\xi + w) + a_0. \end{aligned} \tag{114}$$

Moreover, the BYY harmony learning pushes $(1 - \alpha_j^z)\alpha_j^z \to 0$ if the dimension y_j^z is extra, which is thus discarded during learning. If the mapping $\xi_t \to y_t \to x_t$ improves the standard logistic regression by Eq. (108) and there keep more than one hidden dimensions, we infer that there are contributions more than the one from Eq. (108), which is an indicator to observe whether there is interaction between two SNPs simultaneously.

6.2 Classification versus testing p-value

In the GWA study, a common practice is testing whether the following null hypothesis H_0 is rejected:

$$H_0:\ \text{this SNP is irrelevant to the disease.} \tag{115}$$

The test bases on a test statistic s calculated from samples of the genotypes of this SNP. There is a "null value" s_0 (typically, $s_0 = 0$) such that a value of s close to s_0 presents the strongest evidence in favor of the null hypothesis H_0, whereas a value of s far from s_0 presents the strongest evidence against H_0. One key issue of considering a test statistic is that we must be able to determine its distribution $q(s|H_0)$ under the null hypothesis, which allows us to calculate the so called p-value from a sample set, as shown in Fig. 13(b). If H_0 is valid, samples of the genotypes of this SNP contain no information about the disease, the value $\hat{s}$ calculated from this sample set is not large, and the corresponding p-value is not small. Thus, H_0 is not rejected. On the contrary, rejecting H_0 indicates that samples of this SNP contain some information about the disease.

Practically, the null hypothesis H_0 by Eq. (115) is not easy to implement. There could be different choices for considering an implementable alternative. One typical example could be as follows:

$$H_0:\ p_{S_1} \text{ is not different from } p_{S_0}, \tag{116}$$

where p_{S_ℓ}, $\ell = 0, 1$, is a sample distribution of the genotypes of this SNP obtained from a set S_1 of case samples with the disease and a set S_0 of control samples without the disease, respectively. Still, there are different choices for getting a statistic and conducting its corresponding testing. One example is the Pearson's chi-squared test.

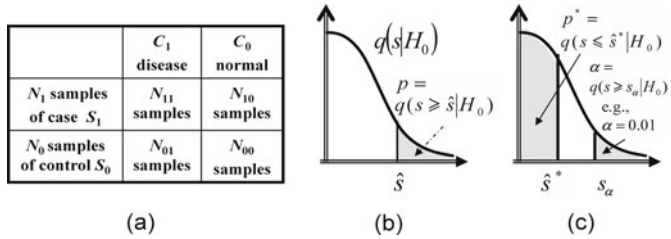

Fig. 13 Classification versus testing statistic p-value

Instead of testing H_0 by Eq. (116), we may classify each sample into a confusion table T as shown in Fig. 13(a), e.g., by the following Bayes classifier:

$$y_t \text{ is classified to } C_1 \text{ if } \alpha_1 p_{S_1}(y_t) > \alpha_0 p_{S_0}(y_t), \tag{117}$$

where y_t is the genotype of one sample about the SNP in consideration, α_0, α_1 are proportions that can be simply estimated by $\alpha_0 = N_0/(N_0 + N_1)$ and $\alpha_1 = N_1/(N_0 + N_1)$. The discrepancy of the resulted confusion table from the desired result $\mathrm{diag}[N_1, N_2]$ describes the performance of classification, which is a common practice in the literature of pattern recognition.

Being different from testing H_0 by Eq. (116) that provides a collective decision on whether H_0 should be rejected, the confusion table by Eq. (117), shortly denoted by T_B, comes from a decision to each individual sample on whether or not this sample is associated with this disease. Moreover, this T_B also serves as a reference or a baseline for making comparisons with confusion tables obtained by different classifiers, e.g., by using either one of three techniques in Sect. 6.1, namely, a logistic regression by Eq. (108), a simplified NFA learning by Eq. (109), and a BYY harmony learning based three layer networks by Eq. (113) or Eq. (114).

We consider testing H_0 by Eq. (115) based on a confusion table T. When H_0 by Eq. (115) is valid, we have

$$\begin{aligned} H_0: &N_{01} = N_{00},\ N_{11} = N_{10}, \\ &\text{or equivalently } T = T_0, \\ &\text{where}\quad T_0 = \begin{pmatrix} N_1 & 0 \\ 0 & N_2 \end{pmatrix} \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}, \end{aligned} \tag{118}$$

from which we may either use Pearson's chi-squared test and Fisher's combined probability test, or develop a new statistic s to test the discrepancy of T from T_0.

However, rejecting H_0 by Eq. (118) is not enough for us to reject H_0 by Eq. (115), since a rejection of Eq. (118) may be not due to that the information contained in the set S_1 of case samples is discriminative from the one in the set S_0 of control samples, but come from a system bias of the classifier that outputs this confusion table T. Taking such a system bias in consideration, we train a pair of classifiers as follows:

- Detector $C(\theta)$: a classifier trained from a set S_0 of control samples and a set S_1 of case samples, with the resulted confusion table denoted by T;
- Reference $C(\theta^*)$: a classifier trained from the above same set S_0 of control samples but with the set S_1 replaced by another set S_1^* of control samples, with the resulted confusion table denoted by T^*.

The detector and reference share a same statistical model $C(\theta)$. This $C(\theta)$ includes a parametric model in a conventional sense. Moreover, a nonparametric model may be expressed in a form of $C(\theta)$, e.g., for p_{S_0} and p_{S_1} given by Eq. (117) with each specific ℓ and each specific y, $p_{S_\ell}(y)$ can be regarded as one unknown parameter that is estimated by sample frequency. All such parameters form θ that is estimated from S_0 and S_1. Similarly, the above θ^* comes from replacing p_{S_1} by $p_{S_1^*}$ that is estimated from S_1^*. Accordingly, we get a confusion table T_B by Eq. (117) and its counterpart T_B^* by replacing p_{S_1} in Eq. (117) with $p_{S_1^*}$.

Moreover, to estimate the unknown parameters, $C(\theta)$ and $C(\theta^*)$ share a same learning principle or an error function to be minimized. The difference lays in the specified values of θ and θ^* that comes from different sample sets. E.g., either p_{S_1} or $p_{S_1^*}$ is estimated by frequency counting, which actually follows from the maximum likelihood learning principle.

Instead of testing H_0 by Eq. (118) merely on one confusion table T, we make the following paired testing:

$$\begin{aligned}&\text{Step } T^*: \text{ testing } H_0 \text{ by Eq. (118) based on } T^* \text{ to check}\\ &\qquad\text{if } H_0 \text{ is not rejected at a significant level } \alpha\\ &\qquad\text{with its statistic } \hat{s} \ll s_\alpha \text{ and a large value}\\ &\qquad p^* \text{ as shown in Fig. 13(c); otherwise quit.}\\ &\text{Step } T: \text{ testing } H_0 \text{ by Eq. (118) based on } T \text{ to check}\\ &\qquad\text{if } H_0 \text{ is rejected at a significant level } \alpha\\ &\qquad\text{with a small } p \text{ value; otherwise fail.}\end{aligned} \tag{119}$$

Step T^* checks whether the statistical model and the learning principle makes a system bias to cause the null hypothesis rejected even when samples contains no information about the disease. As illustrated in Fig. 13(c), a significant level α specifies a boundary point s_α, we prefer the statistic $\hat{s} \ll s_\alpha$ such that the corresponding p^* value is large, which indicates having a large probability that the discrepancy of T from T_0 is not too large to incur a big system bias. However, when $\hat{s} > s_\alpha$, we reject H_0 and quit with failure because we are unable to judge whether a rejection of H_0 at the next step comes from this system bias or some discriminative information contained in the set S_1 of case samples. In such a case, we may either increase the samples sizes of S_0 and S_1^* to re-train the classifier or select a better statistical model and a better learning principle.

Getting a success at Step T^*, we move to test Step T. We prefer that H_0 is rejected at the given significant level with a p value as smaller as possible, together with $\hat{s} \ll s_\alpha$ and a large value of p^* at Step T^*.

Efforts are also suggested to put on comparisons with the implementation of Eq. (119) by $T = T_B$ and $T^* = T_B^*$, which serves as a baseline or a bridge. On one hand, it links to those confusion tables obtained by different classifiers. On the other hand, it links to the null hypothesis H_0 by Eq. (116). Accordingly, we may observe whether a p_{S_0}, p_{S_1} based null hypothesis can be further improved via seeking a better classifier.

The last but not least, the classification associated with the test H_0 by Eq. (115) is a special type of two class problem, with the "case" class in a major consideration while the "control" class distributed flatly as a background. That is, it is actually an one-class decision problem, which motivates us to prefer a classifier designed particularly for such an one-class decision. Taking the simplified NFA learning in Sect. 6.1 as an example, we may modify Eq. (109) with $\eta(\hat{\xi}_t^{(i)})$ replaced by the following one:

$$\eta(u) = \frac{2}{1+e^{u^2}}, \tag{120}$$

which puts the class $\xi = 0$ as the one-class to be focused.

6.3 SNP analysis versus exome sequencing analysis

Typically, the SNP analysis considers statistical test on each SNP, and each sample of the SNP is a label or discrete number that represents the genotype of the SNP. Accordingly, the computation of p-value involves merely a summation or integral of one variate. Recent efforts on SNP based analysis further proceed to exome sequencing analysis, as shown in Fig. 14(a). Each exon sequence may contain multiple SNPs plus other information. It is no longer enough to use one label to represent an exon

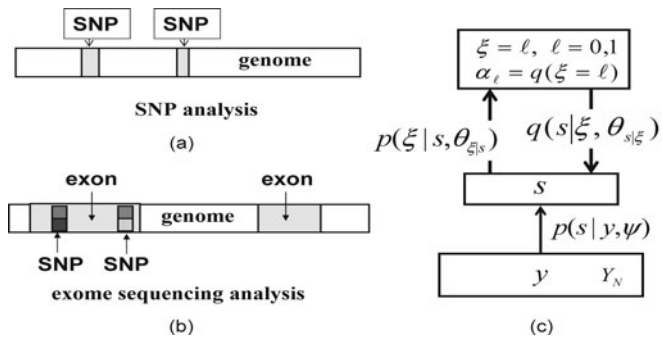

Fig. 14 Exome sequencing analysis and dimension reduction

sequence. Thus, a sample vector of multiple elements is used to encode information about not only each of SNPs but also interactions among these SNPs and across the rest parts of the exon. Accordingly, the counterpart of the null hypothesis H_0 by Eq. (115) becomes

$$H_0 : \text{this exon sequence contains no information about the disease.} \quad (121)$$

However, it becomes more difficult to get the distribution $q(s|H_0)$ in Fig. 13(b). In the sequel, this difficult is tackled along two directions.

One direction is getting confusion tables T and T^* by one of classifiers in Sect. 6.1 or one of other classifiers with a good generalization ability. Then, we conduct the null hypothesis H_0 by Eq. (121) with help of the implementation of Eq. (119). Summarized below are some guidelines on further investigations:

(a) For an exon sequence that contains one SNP plus other features, we get confusion tables T and T^* by using a classifier on a sample set $Y_N = \{y_t\}$, where each y_t is a vector consisting of a number of features extracted from the exon sequence in consideration. Then, the implementation of Eq. (119) is made on these T and T^* in comparison with its implementation on $T = T_B$ and $T^* = T_B^*$ as a baseline. Also, we may test a null hypothesis H_0 that T does differ from T_B. Accordingly, we can observe whether a p_{S_0}, p_{S_1} based null hypothesis by Eq. (116) can be further improved with help of making such a classification on samples from the exon sequence.

(b) For an exon sequence that contains a number of $SNP_j,\ j = 1, 2, \ldots, \kappa$, plus other features, we may get one classifier same as in the above (a) to obtain T and T^*. Then, we test H_0 by Eq. (121) with help of the implementation of Eq. (119) on the obtained T and T^*, in comparison with
   - testing H_0 by Eq. (116) on each of $SNP_j,\ j = 1, 2, \ldots, \kappa,$, we combine the p-values of individual tests (e.g., by Fisher's combined probability test).
   - Getting $T_B^{(j)}$ by Eq. (117) on each of $SNP_j,\ j = 1, 2, \ldots, \kappa$, individually, we get a combination

$$T_B^{\text{com}} = F(T_B^{(j)},\ j = 1, 2, \ldots, \kappa), \quad (122)$$

     where $F(\cdot)$ is a combining rule, a simplest example is taking the average, and other possible combining rules are referred to Ref. [94]. Then, we make the implementation of Eq. (119) on $T = T_B$ and $T^* = T_B^*$.
   - Getting sample distributions $p_{S_0}^{(j)}, p_{S_1}^{(j)}$ on each of $SNP_j,\ j = 1, 2, \ldots, \kappa$, individually, we develop a classifier by considering all sample distributions jointly and get confusion tables T_B^{com} and $T_B^{\text{com}*}$. Then, we make the implementation of Eq. (119) on $T = T_B^{\text{com}}$ and $T^* = T_B^{\text{com}*}$.

The other direction is making a dimension reduction from a high dimensional vector y to a low dimension sample s that is either directly used as a statistic or easy to form a statistic. As shown in Fig. 14(a), an SNP analysis can be regarded as a degenerated case that y reduces into a discrete number representing the SNP's genotype. To map y into s, we may use one of existing dimension reduction methods as a pre-processing. However, a separated pre-processing may lose useful information for a subsequent classification.

Instead, we suggest a dimension reduction $y \to s$ by $p(s|y, \psi)$ to be learned together with the Ying-Yang system as shown in Fig. 14(c). From Eq. (61), we have

$$\begin{aligned} &H_t(p||q, \theta) \\ &\quad = \sum_{\xi} \int p(\xi|s, \theta_{\xi|s}) p(s|\psi) \ln [q(s|\xi, \theta_{s|\xi}) q(\xi) q(\theta)] \mathrm{d}s, \\ &p(s|\psi) = \int p(y|Y_N) p(s|y, \psi) \mathrm{d}y. \end{aligned} \quad (123)$$

Typically, from $p(y|Y_N) = \frac{1}{N} \sum_t \delta(y - y_t)$ and $p(s|y, \psi) = \delta(s - Wy - c)$, we have

$$\begin{aligned} &H_t(p||q, \theta) \\ &\quad = \frac{1}{N} \sum_{\xi} p(\xi|Wy_t + c, \theta) \ln [q(Wy_t + c|\xi, \theta) q(\xi) q(\theta)], \\ &\text{with } p(s|\psi) = \frac{1}{N} \sum_t \delta(s - Wy_t - c). \end{aligned} \quad (124)$$

Approximately, we may regard that $s_t = Wy_t + c$ comes from a Gaussian, which is justified when the dimension of y_t is high and the elements of y_t are mutually independent. When H_0 by Eq. (121) is valid, both case and control samples of y_t come from a same distribution, and thus both case and control samples of s_t belong to a same Gaussian distribution. A practical implementation is considering the following null hypothesis:

$$H_0 :\ \mu_0 = \mu_1, \quad (125)$$

where μ_0, μ_1 are respectively the sample means of the case and the control samples of s_t, subject to either a same variance or different variances. Also, we can make a comparison with Fisher's linear discriminant analysis.

The unknown parameters $\theta = \{W, c, \mu_\ell, \sigma_\ell^2, \alpha_\ell, \ell = 0, 1\}$ are learned via maximizing $H_t(p||q, \theta)$ by Eq. (124), which is further simplified into

$$\begin{aligned} &H_t(p||q, \theta) = \sum_{\ell} p(\ell|s_t, \theta) \ln [G(s_t|\mu_\ell, \sigma_\ell^2) \alpha_\ell q(\theta)], \\ &\text{subject to } s_t = Wy_t + c, \end{aligned} \quad (126)$$

from which we obtain the gradient $\nabla_\varphi H_t(p||q, \theta)$ and develop a semi-supervised learning or supervised learning algorithm, following the developments made in Sect. 2.1.4 and Eq. (99) in Sect. 5.1.4.

Similar to the consideration made at the end of Sect. 6.1, we may also handle the above problem of one-class decision via enforcing σ_0^2 upper-bounded by a small value while σ_1^2 lower-bounded by a large value, with $\sigma_1^2 \gg \sigma_0^2$.

After learning, we get $S_N = \{s_t\}$ from $Y_N = \{y_t\}$ by $s_t = Wy_t + c$ and obtain confusion tables T and T^* by $p(\ell|s_t, \theta)$. Then, we test H_0 by Eq. (121) with help of the implementation of Eq. (119) on these T and T^*. Also, we may test a null hypothesis H_0 that T does differ from T_B. Moreover, we may make a comparative study with Fisher's linear discriminant analysis.

6.4 Gene transcriptional regulation

We move to modeling gene transcriptional regulation by a noisy BFA, that is, a simplified NFA learning at a special case that $\ell = 1, 2$, for which we have $q(y^z|\alpha^z)$ given by Eqs. (111) and (110). Moreover, we have y_j^r from $q(y_j^r|\theta_{j,y_j^z})$ by Eq. (79) with y_j^z in place of $y_{j\ell}^z$. At a special setting that $\mu_{j,1} = 1, \mu_{j,0} = 0$, and $\lambda_{j,1} = \lambda_{j,0} = 0$, we encounter actually a BFA.

With $\lambda_{j,1}, \lambda_{j,0}$ becoming unknown parameters, we are further lead to a noisy BFA when y_j^r that takes binary values $0, 1$ becomes to take real vales from Gaussians centered at 0 and 1, respectively.

Such a noisy BFA is suggested to model gene transcriptional regulation, which leads to further modifications of networks component analysis (NCA) [20,21]. Specifically, different settings of $\mu_{j,1}, \mu_{j,0}$, and $\lambda_{j,1}, \lambda_{j,0}$ lead to several scenarios as follows:

- We modify a BFA by setting some elements of A to be 0, that is, NCA is modified with a new feature that $y_j^r = 1$ indicates a transcription factor (TF) activated while $y_j^r = 0$ indicates that there is no TF activation. When $\mu_{j,1} = 1, \mu_{j,0} = 0$ and $\lambda_{j,1} = \lambda_{j,0} = 0$, it follows that $y_j^z = 1$ implies $y_j^r = 1$, while $y_j^z = 0$ implies $y_j^r = 0$.
- We further modify this semi-blind BFA by relaxing $\mu_{j,1}$ and $\lambda_{j,1}$ to be free to take unknown parameters, while we still set $\mu_{j,0} = 0$ and $\lambda_{j,0} = 0$ by which y_j^r from $G(y_j^r|0, 0)$ indicates $y_j^r = 0$ (no TF activation). On the contrary, y_j^r from $G(y_j^r|\mu_{j,1}, \lambda_{j,1})$ indicates a TF activation with its strength varying randomly around $\mu_{j,1}$ with a variation described by $\lambda_{j,1}$. During learning, one scenario is that each y_j^z may be known to take a label $y_j^z = 1$ or $y_j^z = 0$ according to whether a TF is known in binding to this gene. The other scenario is estimating a unknown y_j^z to check whether there is a binding.
- This semi-blind BFA can be regarded as modified from a standard BFA simply with y_t replaced by

$$y_t = y_t^r \circ y_t^z = [y_1^r y_1^z, y_2^r y_2^z, \ldots, y_m^r y_m^z]^{\mathrm{T}}, \quad (127)$$

and accordingly a learning algorithm can be obtained from a standard BFA algorithm with help of the following three-step-alternation:

1) Given y_t^z, A, and the statistics of noise e, we estimate y_t^r by a weighted BFA $x_t = A^z y_t^r + e_t$ with $A^z = A\ \mathrm{diag}[y_1^z, y_2^z, \ldots, y_m^z]$;
2) Given y_t^r, A, and the statistics of noise e, we estimate y_t^z by a weighted BFA $x_t = A^r y_t^z + e_t$ with $A^r = A\ \mathrm{diag}[y_1^r, y_2^r, \ldots, y_m^r]$;
3) Get y_t by Eq. (127) from given y_t^r and y_t^z, we estimate A, and the statistics of noise e from a standard BFA $x_t = Ay_t + e_t$.

- A further extension considers that elements of A switches between different modes. One example is shown in Fig. 4 (particularly Eq. (100)) of Ref. [3], where A is replaced by $A \circ L = [a_{ij}\ell_{ij}]$ with $\ell_{ij} = 1$ or 0. Learning $x_t = (A \circ L)y_t + e_t$ involves optimizing a binary matrix L per implementation of the Yang-step.
- To avoid searching a binary matrix, we return to consider $x_t = Ay_t + e_t$ with help of a prior $q(\theta)$ in Eq. (98) that includes $q(A, L)$, typically with

$$\begin{aligned} q(a_{ij}|\ell_{ij} = 1) &= G(a_{ij}|\mu_{ij}^a, \gamma_{ij,1}^a), \\ q(a_{ij}|\ell_{ij} = 0) &= G(a_{ij}|0, \gamma_{ij,0}^a), \end{aligned} \quad (128)$$

where a_{ij} coming from $G(a_{ij}|0, \gamma_{ij,0}^a = 0)$ is equivalent to setting $a_{ij} = 0$ probabilistically. Alternatively, relaxing $\gamma_{ij,0}^a$ to take a small value or an unknown value to be estimated during learning, a hard switching off $a_{ij} = 0$ is relaxed to taking a small value randomly as if some background noises. On the contrary, a_{ij} from $G(a_{ij}|\mu_{ij}^a, \gamma_{ij,1}^a)$ indicates the connectivity with its strength varying randomly around μ_{ij}^a with a variance $\gamma_{ij,1}^a$. Moreover, the connectivity can be further examined via getting $p(\ell_{ij}|x_t)$ and $p(\ell_{ij}|x_t, A)$.

7 Closing outlines

This paper provides a further supplementary of Refs. [1] and [3]. From different aspects and with different focuses, the three sister papers jointly provide a systematic overview and also a tutorial on the BYY harmony learning. Generally speaking, Ref. [1] serves as a core reading on fundamentals and important topics, while this paper supplements Ref. [1] with the following motivations:

- Important topics are elaborated systematically in a way that those already addressed in Ref. [1] are briefly outlined or re-organized (if need), with unclear points clarified, some variants provided, and certain missing issues further addressed.
- Several new topics have been added and addressed in details, while they were untouched in Ref. [1].
- A number of challenging issues and further topics are discussed for further investigation.

- Taking Gaussian mixture as an example, an easy understanding tutorial in a bird's-eye view is provided in comparison with typical algorithms.

In Ref. [3], further insights are provided on a family of BYY systems that are set up by specific building units, with each building unit featured by a co-dimensional matrix pair (co-dim matrix pair). The common rank m of each matrix pair is an intrinsic dimension that could be estimated by the information from each of the two matrices, which leads to improved learning performance with not only refined model selection criteria but also a modified mechanism that coordinates automatic model selection and sparse learning. A number of typical latent variable based models are covered, with the corresponding learning algorithms developed.

For clarity and completeness, the major topics introduced in Refs. [1] and [3], and this paper are summarized as follows.

• Topics about statistical learning in general

1) *Regularization, sparse learning, and model selection*, which are mainly introduced and discussed in Sect. 2.1 (see page 286) of Ref. [1], with a supplementary discussion made in Sect. 4.4.2 (d) of this paper. Particularly, the common point and difference between automatic model selection and sparse learning as well as their coordination are further addressed via the co-dim matrix pair based models in Sect. 2.2 of Ref. [3], especially after Eq. (34).
2) *Two stage implementation, stepwise implementation, automatic model selection*, which are mainly introduced and discussed in Sect. 2.1 (see page 287) of Ref. [1]. Moreover, detailed formulations of Eq. (4) of Ref. [1] are further given by Eqs. (35) and (36) of Ref. [3] for a co-dim matrix pair, and by Eq. (42) of this paper for a Gaussian mixture.
3) *IBC prior*, which bases on a belief that a prior consists of canceling out a system bias by a IBC prior and then adding an informative prior. In Ref. [1], it is introduced by the last part of Sect. 4.2 on page 303. Details are referred to Sect. 3.4.3 in Ref. [41].

• Topics about BYY system

1) *A modern Yin-Yang viewpoint*, which is introduced in Ref. [1] (see the first two paragraphs of Sect. 4.1 and also Appendix B1). In this paper, it is further outlined by the first two paragraphs of Sect. 3.1.1.
2) *Least complexity principle for Ying design*, featured with designing $q(Y)$ in a least redundancy principle and designing $q(X|R)$ in a divide-conquer principle. Further details are referred to Sect. 4.2 (see page 302) in Ref. [1], and Sect. 3.2.1 of this paper.
3) *Variety preservation principle for Yang design*, which is introduced in Sect. 4.2 (see pages 302 and 303) of Ref. [1]. Further details are provided in Sect. 3.2.2 of this paper. Not only $\Pi^q_{Y|X}, \Pi_\theta$ in Eq. (31) of Ref. [1] is refined into $\Pi^q_{Y|X,\theta}$ and $\Pi^q_{\theta|X}$ in Eq. (36), but also another factorization of Yang machine is considered with its corresponding design given by Eq. (43).
4) *Ying-based model selection versus Yang-based regularization*, which is introduced in Sect. 2.2 (see page 280) of Ref. [1]. In this paper, it is also outlined by Sect. 3.3.1, with further discussions on coordination within Ying and within Yang.
5) *Unsupervised vs semi-supervised learning*, which is introduced in Sect. 4.4 (see pages 306–308) of Ref. [1], where two typical types of BYY supervised learning are reviewed. In this paper, Sect. 3.3.2 further argues that the BYY system acts as a unified framework to accommodate unsupervised, supervised, and semi-supervised learning all in one formulation.
6) *Semi-blind learning* In Ref. [3], Sect. 4.3 presents a general formulation, together with a semi-blind learning BFA for a specific linear regression task in the last part of Sect. 3.2. In this paper, not only this semi-blind learning BFA has been applied to modeling gene transcriptional regulation in Sect. 6.4 with a three-step alternating implementation, but also a semi-blind learning NFA has been proposed for genome-wide association study in Sect. 6.1.
7) *Co-dim matrix pair and post bi-linear matrix based BYY system*, which is introduced and addressed by Sect. 2 of Ref. [3].
8) *Hierarchy of co-dim matrix pairs*, which is introduced and addressed by Sect. 4.1 of Ref. [3].

• Topics about best harmony learning

1) *Bi-entity proximity: equivalent vs different* As introduced in Appendix A and Fig. A1 of Ref. [1], measures of bi-entity proximity come from either a least difference perspective or a best agreement perspective. Two perspectives are usually different though becoming equivalent in some special cases. Actually, $H_\mu(P||Q)$ evolves from the former while the Kullback divergence evolves from the latter.
2) *Harmony functional: triple-relation vs bi-relation* Radon-Nikodym derivative based harmony functional $H_\mu(P||Q)$ is a triple-relation that not only includes Kullback divergence and Shannon entropy (each is a bi-relation) as special cases but also differs from the minimum cross entropy (MCE), which is introduced in Sect. 4.1 (see page 299) in Ref. [1]. In this paper, this issue is systematically re-elaborated in Sect. 4.1 and Fig. 5, with $f(r)$ further specified to clarify some unclear issue, and also with further explanation on why the name of harmony functional

is adopted.

3) *Unidirectional learning vs bidirectional learning* As discussed in Sect. 4.1 of this paper, there are many choices for extending bi-entity proximity to a BYY system, from both a unidirectional perspective and a bidirectional perspective, among which $H_\mu(P||Q)$ is argued as the best.
4) *Ying-Yang best harmony* It seeks a best Ying-Yang matching with a least complexity in term of the system entropy, as introduced in Sect. 4.1 of Ref. [1] (especially the third paragraph on page 299), and also outlined in Sect. 4.2.3 of this paper (see the first aspect of observation on the novelty).
5) *Ying-Yang tacit matching* Ying-Yang best harmony also means that Ying-Yang seeks a best agreement in a most tacit manner via a least amount of information communication from Yang, as introduced in Sect. 4.1 of Ref. [1] (especially the paragraph around Eq. (24) on page 299), and also outlined in Sect. 4.2.3 of this paper (see the third aspect of observation on the novelty).
6) *$q(Y)$ in a scale sensitive position*, which is addressed in Sect. 2.2 and Fig. 5 of Ref. [1]. In this paper, it is further outlined in Sect. 4.2.3 (see the second aspect of observation on the novelty).
7) *BYY best harmony versus discriminative training*, which is addressed in this paper at the ending part of Sect. 4.2.1.
8) *Manifold shrinking*, which is introduced in Fig. 10 on page 300 of Ref. [1], and further addressed in this paper by the last part of Sect. 4.3.2.
9) *RPCL like gradient flow*, which is introduced around Eq. (25) on page 300 of Ref. [1], and a variant is given by Eq. (22) in Ref. [62].
10) *Three-level encoding based optimal communication*, which is introduced in Sect. II(C), Sect. II(E), and Fig. 3 in Ref. [4], and is outlined by the first paragraph on page 302 of Ref. [1].
11) *Hierarchical learning and bottom-up decoupling* Minimizing $KL(P||Q)$ has a bottom-up decoupling nature, while maximizing $H_\mu(P||Q)$ does not have, as addressed by Sect. 5.2 of Ref. [1] (especially the third paragraph on page 299) and also outlined in Sect. 4.2.3 of this paper (see the third aspect of observation on the novelty).
12) *Temporal decoupling nature* As addressed at the end of Sect. 5.2 in Ref. [1], maximizing $H_\mu(P||Q)$ has a temporal decoupling nature, while minimizing $KL(P||Q)$ does not have.
13) *A unified framework of statistical learning* Maximizing $H_\mu(P||Q)$ (including minimizing $KL(P||Q)$ as a special case) acts as a general framework that unifies typical learning methods, which is addressed by the second part of Appendix A and Fig. A2 in Ref. [1], and also briefly outlined in Sects. 4.2.2 and 4.2.3 of this paper.

As a whole, the above list extends the nine aspects summarized at the end of Sect. 4.1 in Ref. [1], not only on the novelty and favorable natures of BYY best harmony, but also on its relation to BYY best matching.

- **Topics about BYY learning implementation**

1) *Apex approximation* An integral or summation is approximated around apex zone, and one key technique is given by Eq. (35) in Sect. 4.3 of Ref. [1] (also see Sect. 2.3 in Ref. [3] and Sect. 4.3.1 in this paper), while a summation is approximated by a sum over an apex-zone centered around the peak in one or a few bits, e.g., Eq. (20) in Ref. [1] and Eq. (111) in this paper.
2) *Alternative maximization* The key point is alternatively updating unknowns in Yang with Ying fixed and unknowns in Ying with Yang fixed, see the end part of Sect. 4.3 in Ref. [1]. Considering unknowns hierarchically, the alternation consists of multi-stages hierarchically, see Eq. (43) in Ref. [3] as well as Eq. (18) and Sect. 4.3.1 of this paper.
3) *Partition of priors* Priors are divided into an integrable part and a non-integrable part (see Eq. (42) in Ref. [3]) or alternatively into a part of hyper-parameters and a part of no hyper-parameters (see Eq. (70) in this paper).
4) *Balanced operation* As discussed in Sect. 4.3.3 of this paper, learning implementation should balance learning operation on each part of unknowns to avoid getting trapped at a local maximum. Readers are further referred to Sects. 2.3 and 3.3 in Ref. [1].

- **Exemplar learning tasks and algorithms**

1) *Gaussian mixture* A tutorial on unsupervised learning algorithms is introduced in Sect. 3.1 and Fig. 7 of Ref. [1], implementing the BYY harmony learning in comparison with the EM algorithm for maximizing the likelihood, RPCL learning, κ-MAP EM learning, WTA-BYY harmony, where all the algorithms are summarized in a unified Ying-Yang alternation procedure with major parts in a same expression while differences characterized by few options in some subroutines. In Sect. 3.1 of Ref. [3], extension is made to a supervised learning variant. In Sect. 2.1.5 of this paper, a further extension is made to semi-supervised learning by Eq. (14).
2) *Factor analysis* The BYY harmony learning algorithm is also given for the FA in Sect. 3.2 and Fig. 7 of Ref. [1], in comparison with the EM algorithm for maximizing the likelihood. Moreover, the co-dim matrix pair featured FA is proposed by Eq. (62) in Ref. [3] for improving automatic model selection on the number of factors. Being different from the

traditional parameterization of FA (shortly FA-a), another parameterization (shortly FA-b) is considered for the BYY harmony learning, see Item 9.4 in Ref. [26] and Sect. 3 in Ref. [33]. Though FA-a and FA-b are equivalent in term of maximizing the likelihood, extensive empirical experiments in Ref. [34] has shown that the BYY harmony learning and VB perform reliably and robustly better on FA-b than on FA-a, while BYY further outperforms VB considerably, especially on FA-b.

3) *NFA, BFA, and three-layer networks* The learning algorithm given in Sect. 3.2 and Fig. 7 of Ref. [1] is actually all in one formulation. Also, the co-dim matrix pair featured by Eq. (62) in Ref. [3] can be approximately used for BFA and NFA, as addressed by the paragraphs from Eq. (62) to Eq. (64) in Ref. [3]. Partitioning the input of BFA into two parts leads to a BYY harmony learning implementation for a classic three-layer network, as outlined by the paragraphs from Eq. (48) to Eq. (53) in Ref. [1].
4) *Mode-switching factor analysis, semi-blind learning, and semi-blind FA* As shown in Sect. 5.1.1 and Fig. 7, mode-switching factor analysis is a general formulation that includes Gaussian FA, BFA, NFA, and their variants of semi-blind learning.
5) *Manifold learning* As addressed by the paragraphs from Eq. (65) to Eq. (68) in Ref. [3], the popular graph Laplacian based manifold learning can be equivalently reformulated as a co-dim matrix pair based BYY harmony learning with automatic model selection and learning regularization. It follows from Eq. (66) in Ref. [3] that this manifold learning can be regarded as an extension of FA-a. In this paper, a further improvement is suggested to its counterpart of FA-b with Eq. (66) in Ref. [3] replaced by Eq. (107).
6) *LFA and SBF* In Sect. 3.2 and Fig. 8 of Ref. [1], the BYY harmony learning algorithm is developed for implementing local FA in comparison with the EM algorithm. In Sect. 4.2 of Ref. [3], a de-noise local FA is further proposed and then extended with each subspace supported by a cascaded linear regression, with the number of subspaces and the dimension of each subspace determined during the BYY harmony learning.
7) *Mixture of experts and RBF networks* As illustrated by the Box ⑨ in Fig. 11 of Ref. [1], we are lead to RBF networks and alternative mixture of experts. Further details are referred to a recent overview in Ref. [38].
8) *TFA, temporal BFA, and temporal NFA* Taking temporal dependence in consideration, FA is extended to TFA. All in one formulation in Sect. 3.2 and Fig. 7 of Ref. [1] also cover a first order approximation of the BYY harmony learning on TFA [40,61]. Without approximation, the implementing techniques for the BYY harmony learning on TFA are addressed in Sect. 5.1 and especially Fig. 13 of Ref. [1]. In Sect. 4.2 and especially Eqs. (92) and (93) in Ref. [3], the co-dim matrix pairing nature has been generalized to TFA and the state space model, and a double loop learning procedure is proposed, sharing the nature of automatic model selection and sparse learning. Also, TFA has been applied to HRRP data for radar object recognition [88].
9) *HMM model and HMM gated TFA* In Sect. 5.3 of Ref. [1], learning algorithms are provided not only for implementing the BYY harmony learning with automatic model selection on hidden states (see Fig. 12), but also for discriminative learning of multiple HMM models. In Sect. 5.2.2 of this paper, HMM model and TFA models are combined to form a HMM gated TFA for modeling long range across-frame temporal dependence, with each hidden state using one TFA model for stationary temporal dependence within each segment.
10) *Hierarchical Gaussian mixture* In Sect. 5.1 and Fig. 12 of Ref. [1], a hierarchical Gaussian mixture is addressed with a BYY harmony learning provided, in comparison with the EM algorithm. Taking a two-level hierarchical Gaussian mixture as an example, the learning algorithm is extended to supervised learning and semi-supervised learning in Sect. 5.3.1 and Fig. 11 of this paper.
11) *Graph matching, covariance decomposition, and data-covariance co-decomposition* In Sect. 3.3 of Ref. [3], attributed graph matching is formulated as a decomposition of covariance, which is implemented by optimizing one of two cost functions subject to an orthostochastic matrix constraint. Moreover, it is further extended to a co-decomposition of data and covariance for a better performance.
12) *PPI network and network alignment* A BYY harmony learning based bi-clustering algorithm has been developed for PPI network partitioning with favorable performances in comparison with several well known clustering algorithms [95]. Further improvements are suggested from the co-dim matrix pair perspective in Ref. [3]. Moreover, network alignment is taken in consideration via graph matching from a perspective of data-covariance co-decomposition with help of the BYY harmony learning, which provides a potential formulation for integrating data types across several domains.
13) *Gene transcriptional regulation* In Sect. 3.3 of Ref. [3], past studies have been summarized in three streams of advances, and further progresses are made in help with a co-dim matrix pair perspective of the BYY harmony learning, especially a general

formulation for semi-blind learning and its extension for temporal modeling. In Sect. 6.4 of this paper, a noisy BFA with a three-step alternation procedure is suggested to improve networks component analysis [20,21] for gene transcriptional regulation.

14) *Genome-wide association study* In Sect. 6.1 of this paper, a formation of semi-supervised learning is suggested for regression analysis with automatic selection on variables by which we analyze the relations between a set of SNPs and multiple complex traits in GWA study.

15) *Exome sequencing analysis* In Sect. 6.3 of this paper, our efforts proceed to exome sequencing analysis along two directions. One is getting a confusion table by one of classifiers with a good generalization ability, and testing a null hypothesis from this confusion table by an appropriate statistic. The other direction is making a dimension reduction by learning a BYY system with its Yang pathway as a classifier for getting a confusion table.

The last but not least, ten further topics are listed at the end of Sect. 6 in Ref. [1] for future studies. In this paper, challenge issues and topics for future studies are given in Sects. 3.4 and 4.4, plus additional issues scatted at the ends of subsections of Sects. 5 and 6 as well.

Acknowledgements This work was supported by the General Research Fund from Research Grant Council of Hong Kong (Project No. CUHK418012E), and the National Basic Research Program of China (973 Program) (No. 2009CB825404). The work also came from five years of summer lectures taught by the present author in Peking University, supported by Chang Jiang Scholars Program by Ministry of Education of China and Peking University for Chang Jiang Chair Professorship.

References

1. Xu L. Bayesian Ying-Yang system, best harmony learning, and five action circling. A special issue on Emerging Themes on Information Theory and Bayesian Approach. Frontiers of Electrical and Electronic Engineering in China, 2010, 5(3): 281–328
2. Xu L. Bayesian-Kullback coupled YING-YANG machines: Unified learning and new results on vector quantization. In: Proceedings of the International Conference on Neural Information Processing. 1995, 977–988 (A further version in NIPS8. In: Touretzky D S, et al. eds. Cambridge: MIT Press, 444–450)
3. Xu L. Codimensional matrix pairing perspective of BYY harmony learning: Hierarchy of bilinear systems, joint decomposition of data-covariance, and applications of network biology. A special issue on Machine Learning and Intelligence Science: IScIDE2010 (A). Frontiers of Electrical and Electronic Engineering in China, 2011, 6(1): 86–119
4. Xu L. Advances on BYY harmony learning: Information theoretic perspective, generalized projection geometry, and independent factor autodetermination. IEEE Transactions on Neural Networks, 2004, 15(4): 885–902
5. Xu L. Temporal BYY encoding, Markovian state spaces, and space dimension determination. IEEE Transactions on Neural Networks, 2004, 15(5): 1276–1295
6. Xu L. Bayesian Ying Yang system, best harmony learning, and Gaussian manifold based family. In: Zurada et al. eds. Computational Intelligence: Research Frontiers, WCCI2008 Plenary/Invited Lectures. Lecture Notes in Computer Science, 2008, 5050: 48–78
7. Shi L, Tu S K, Xu L. Learning Gaussian mixture with automatic model selection: A comparative study on three Bayesian related approaches. A special issue on Machine Learning and Intelligence Science: IScIDE2010 (B). Frontiers of Electrical and Electronic Engineering in China, 2011, 6(2): 215–244
8. Shore J. Minimum cross-entropy spectral analysis. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1981, 29(2): 230–237
9. Burg J P, Luenberger D G, Wenger D L. Estimation of structured covariance matrices. Proceedings of the IEEE, 1982, 70(9): 963–974
10. Jaynes E T. Information theory and statistical mechanics. Physical Review, 1957, 106(4): 620–630
11. Schwarz G. Estimating the dimension of a model. Annals of Statistics, 1978, 6(2): 461–464
12. MacKay D J C. A practical Bayesian framework for backpropagation networks. Neural Computation, 1992, 4(3): 448–472
13. Attias H. A variational Bayesian framework for graphical models. Advances in Neural Information Processing Systems, 2000, 12: 209–215
14. McGrory C A, Titterington D M. Variational approximations in Bayesian model selection for finite mixture distributions. Computational Statistics & Data Analysis, 2007, 51(11): 5352–5367
15. Amari S I, Cichocki A, Yang H. A new learning algorithm for blind separation of sources. In: Touretzky D S, Mozer M C, Hasselmo M E, eds. Advances in Neural Information Processing System 8. Cambridge: MIT Press, 1996, 757–763
16. Bell A J, Sejnowski T J. An information-maximization approach to blind separation and blind deconvolution. Neural Computation, 1995, 7(6): 1129–1159
17. Xu L. Independent subspaces. In: Ramón J, Dopico R, Dorado J, Pazos A, eds. Encyclopedia of Artificial Intelligence. Hershey, PA: IGI Global, 2008, 903–912
18. Bahl L, Brown P, de Souza P, Mercer R. Maximum mutual information estimation of hidden Markov model parameters for speech recognition. In: Proceedings of 1986 IEEE International Conference on Acoustics, Speech, and Signal Processing. 1986, 11: 49–52
19. Valtchev V, Odell J J, Woodland P C, Young S J. MMIE training of large vocabulary recognition systems. Speech Communication, 1997, 22(4): 303–314
20. Liao J C, Boscolo R, Yang Y L, Tran L M, Sabatti C, Roychowdhury V P. Network component analysis: Reconstruction of regulatory signals in biological systems. Proceedings

of the National Academy of Sciences of the United States of America, 2003, 100(26): 15522–15527

21. Brynildsen M P, Tran L M, Liao J C. A Gibbs sampler for the identification of gene expression and network connectivity consistency. Bioinformatics, 2006, 22(24): 3040–3046
22. Redner R A, Walker H F. Mixture densities, maximum likelihood, and the EM algorithm. SIAM Review, 1984, 26(2): 195–239
23. Xu L, Jordan M I. On convergence properties of the EM algorithm for Gaussian mixtures. Neural Computation, 1996, 8(1): 129–151
24. Xu L, Krzyzak A, Oja E. Rival penalized competitive learning for clustering analysis, RBF net, and curve detection. IEEE Transactions on Neural Networks, 1993, 4(4): 636–649
25. Xu L. Best harmony, unified RPCL and automated model selection for unsupervised and supervised learning on Gaussian mixtures, three-layer nets and ME-RBF-SVM models. International Journal of Neural Systems, 2001, 11(1): 43–69
26. 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. Springer-Verlag, 1997, 241–274
27. Salah A A, Alpaydin E. Incremental mixtures of factor analyzers. In: Proceedings the 17th International Conference on Pattern Recognition. 2004, 1: 276–279
28. Williams P M. Bayesian regularization and pruning using a Laplace prior. Neural Computation, 1995, 7(1): 117–143
29. Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B: Methodological, 1996, 58(1): 267–288
30. Figueiredo M A F, Jain A K. Unsupervised learning of finite mixture models. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(3): 381–396
31. Corduneanu A, Bishop C M. Variational Bayesian model selection for mixture distributions. In: Richardson T, Jaakkola T, eds. Proceedings of the Eighth International Conference on Artificial Intelligence and Statistics. 2001, 27–34
32. Wallace C S, Dowe D L. Minimum message length and Kolmogorov complexity. Computer Journal, 1999, 42(4): 270–283
33. Xu L. Bayesian Ying Yang system and theory as a unified statistical learning approach (III): Models and algorithms for dependence reduction, data dimension reduction, ICA and supervised learning. In: Wong K M, et al. eds. Theoretical Aspects of Neural Computation: A Multidisciplinary Perspective. Springer-Verlag, 1997, 43–60
34. Tu S K, Xu L. Parameterizations make different model selections: Empirical findings from factor analysis. A special issue on Machine Learning and Intelligence Science: IScIDE2010 (B). Frontiers of Electrical and Electronic Engineering in China, 2011, 6(2): 256–274
35. Xu L. BYY harmony learning, structural RPCL, and topological self-organizing on mixture models. Neural Networks, 2002, (8–9): 1125–1151
36. Ghahramani Z, Beal M. Variational inference for Bayesian mixtures of factor analysers. Advances in neural information processing systems 12. Cambridge, MA: MIT Press, 2000, 449–455
37. Utsugi A, Kumagai T. Bayesian analysis of mixtures of factor analyzers. Neural Computation, 2001, 13(5): 993–1002
38. Xu L. Learning algorithms for RBF functions and subspace based functions. In: Olivas E, et al. eds. Handbook of Research on Machine Learning, Applications and Trends: Algorithms, Methods and Techniques. Hershey, PA: IGI Global, 2009, 60–94
39. Xu L. BYY $\sum$-$\prod$ factor systems and harmony learning. Invited talk. In: Proceedings of International Conference on Neural Information Processing (ICONIP'2000). 2000, 1: 548–558
40. Xu L. BYY harmony learning, independent state space, and generalized APT financial analyses. IEEE Transactions on Neural Networks, 2001, 12(4): 822–849
41. Xu L. A unified perspective and new results on RHT computing, mixture based learning, and multi-learner based problem solving. Pattern Recognition, 2007, 40(8): 2129–2153
42. Xu L. Bayesian Ying Yang learning. In: Zhong N, Liu J, eds. Intelligent Technologies for Information Analysis. Berlin: Springer, 2004, 615–706
43. Barron A, Rissanen J, Yu B. The minimum description length principle in coding and modeling. IEEE Transactions on Information Theory, 1998, 44(6): 2743–2760
44. Bishop C M. Training with noise is equivalent to Tikhonov regularization. Neural Computation, 1995, 7(1): 108–116
45. Zhou Z H. When semi-supervised learning meets ensemble learning. A special issue on Machine Learning and Intelligence Science: IScIDE2010 (A). Frontiers of Electrical and Electronic Engineering in China, 2011, 6(1): 6–16
46. Xu L. RBF nets, mixture experts, and Bayesian Ying-Yang learning. Neurocomputing, 1998, 19(1–3): 223–257
47. Xu L. Independent component analysis and extensions with noise and time: A Bayesian Ying-Yang learning perspective. Neural Information Processing — Letters and Reviews, 2003, 1(1): 1–52
48. Xu L. BYY learning, regularized implementation, and model selection on modular networks with one hidden layer of binary units. Neurocomputing, 2003, 51: 277–301
49. Shilov G E, Gurevich B L. Integral, Measure, and Derivative: A Unified Approach. Silverman R trans. New York: Dover Publications, 1978
50. Povey D, Woodland P C. Minimum phone error and I-smothing for improved discriminative training. In: Proceedings of 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing. 2002, 1: 105–108
51. Juang B H, Katagiri S. Discriminative learning for minimum error classification. IEEE Transactions on Signal Processing, 1992, 40(12): 3043–3054
52. Juang B H, Chou W, Lee C H. Minimum classification error rate methods for speech recognition. IEEE Transactions on Speech and Audio Processing, 1997, 5(3): 257–265
53. Saul L K, Rahim M G. Maximum likelihood and minimum classification error factor analysis for automatic speech recognition. IEEE Transactions on Speech and Audio Processing, 2000, 8(2): 115–125

54. Rissanen J. Modeling by shortest data description. Automatica, 1978, 14(5): 465–471
55. Hinton G E, Dayan P, Frey B J, Neal R M. The "wake-sleep" algorithm for unsupervised neural networks. Science, 1995, 268(5214): 1158–1161
56. Xu L, Oja E, Suen C Y. Modified Hebbian learning for curve and surface fitting. Neural Networks, 1992, 5(3): 441–457
57. Xu L, Krzyzak A, Oja E. A neural net for dual subspace pattern recognition methods. International Journal of Neural Systems, 1991, 2(3): 169–184
58. Hinton G E, Zemel R S. Autoencoders, minimum description length and Helmholtz free energy. In: Cowan J D, Tesauro G, Alspector J, eds. Advances in Neural Information Processing Systems 6. San Mateo: Morgan Kaufmann, 1994, 449–455
59. Xu L, Krzyzak A, Oja E. Unsupervised and supervised classifications by rival penalized competitive learning. In: Proceedings of the 11th International Conference on Pattern Recognition. 1992, I: 672–675
60. Xu L. BYY data smoothing based learning on a small size of samples. In: Proceedings of International Joint Conference on Neural Networks. 1999, 1: 546–551
61. Xu L. Temporal BYY learning for state space approach, hidden Markov model, and blind source separation. IEEE Transactions on Signal Processing, 2000, 48(7): 2132–2144
62. Xu L. Machine learning problems from optimization perspective. Journal of Global Optimization, 2010, 47(3): 369–401
63. Xu L. Bayesian Ying Yang system and theory as a unified statistical learning approach: (II) From unsupervised learning to supervised learning, and temporal modeling. In: Wong K M, King I, Yeung D Y, eds. Proceedings of Theoretical Aspects of Neural Computation: A Multidisciplinary Perspective. 1997, 29–42
64. Xu L. Bayesian-Kullback YING-YANG machines for supervised learning. In: Proceedings of the 1996 World Congress On Neural Networks. San Diego, CA, 1996, 193–200
65. Xu L. Bayesian Kullback Ying-Yang dependence reduction theory. Neurocomputing, 1998, 22(1–3): 81–111
66. Xu L. Bayesian Ying-Yang system and theory as a unified statistical learning approach: (V) Temporal modeling for temporal perception and control. In: Proceedings of the International Conference on Neural Information Processing. 1998, 2: 877–884
67. Xu L. New advances on Bayesian Ying-Yang learning system with Kullback and non-Kullback separation functionals. In: Proceedings of 1997 IEEE-(INNS) Conference on Neural Networks. 1997, 3: 1942–1947
68. Xu L. Bayesian Ying-Yang machine, clustering and number of clusters. Pattern Recognition Letters, 1997, 18(11–13): 1167–1178
69. Xu L. How many clusters?: A YING-YANG machine based theory for a classical open problem in pattern recognition. In: Proceedings of the 1996 IEEE International Conference on Neural Networks. 1996, 3: 1546–1551
70. Xu L. Bayesian Ying-Yang theory for empirical learning, regularization, and model selection: General formulation. In: Proceedings of International Joint Conference on Neural Networks. 1999, 1: 552–557
71. Xu L. Temporal BYY learning and its applications to extended Kalman filtering, hidden Markov model, and sensormotor integration. In: Proceedings of International Joint Conference on Neural Networks. 1999, 2: 949–954
72. Xu L. Temporal factor analysis: Stable-identifiable family, orthogonal flow learning, and automated model selection. In: Proceedings of International Joint Conference on Neural Networks. 2002, 472–476
73. Csiszár I, Tusnády G. Information geometry and alternating minimization procedures. Statistics and Decisions, 1984, (Suppl 1): 205–237
74. Xu L. Temporal Bayesian Ying-Yang dependence reduction, blind source separation and principal independent components. In: Proceedings of International Joint Conference on Neural Networks. 1999, 2: 1071–1076
75. Pang Z H, Tu S K, Su D, Wu X H, Xu L. Discriminative training of GMM-HMM acoustic model by RPCL learning. A special issue on Machine Learning and Intelligence Science: IScIDE2010 (B). Frontiers of Electrical and Electronic Engineering in China, 2011, 6(2): 283–290
76. Amari S, Nagaoka H. Methods of Information Geometry. London, U.K.: Oxford University Press, 2000
77. Belouchrani A, Cardoso J. Maximum likelihood source separation by the expectation maximization technique: deterministic and stochastic implementation. In. Proceedings of NOLTA95. 1995, 49–53
78. McLachlan G J, Krishnan T. The EM Algorithms and Extensions. New York: John Wiley and Sons, 1997
79. Shi L, Tu S K, Xu L. Gene clustering by structural prior based local factor analysis model under Bayesian Ying-Yang harmony learning. In: Proceedings of the 2010 International Conference on Bioinformatics and Biomedicine. 2010, 696–699
80. Tibshirani R. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B: Methodological, 1996, 58(1): 267–288
81. Park M Y, Hastie T. Penalized logistic regression for detecting gene interactions. Biostatistics, 2008, 9(1): 30–50
82. Brown R G, Hwang P Y C. Introduction to Random Signals and Applied Kalman Filtering. 3rd ed. New York: John Wiley and Sons, 1997
83. Roweis S, Ghahramani Z. A unifying review of linear Gaussian models. Neural Computation, 1999, 11(2): 305–345
84. Ghahramani Z, Hinton G E. Variational learning for switching state-space models. Neural Computation, 2000, 12(4): 831–864
85. Shumway R H, Stoffer D S. An approach to time series smoothing and forecasting using the EM algorithm. Journal of Time Series Analysis, 1982, 3(4): 253–264
86. Shumway R H, Stoffer D S. Dynamic linear models with switching. Journal of the American Statistical Association, 1991, 86(415): 763–769
87. Digalakis V, Rohlicek J R, Ostendorf M. ML estimation of a stochastic linear system with the EM algorithm and its application to speech recognition. IEEE Transactions on Speech and Audio Processing, 1993, 1(4): 431–442

88. Wang P H, Shi L, Du L, Liu H W, Xu L, Bao Z. Radar HRRP statistical recognition with temporal factor analysis by automatic Bayesian Ying-Yang harmony learning. A special issue on Machine Learning and Intelligence Science: IScIDE2010 (B). Frontiers of Electrical and Electronic Engineering in China, 2011, 6(2): 300–317
89. Gales M J F, Young S. The application of hidden Markov models in speech recognition. Foundations and Trends in Signal Processing, 2008, 1(3): 195–304
90. Cordell H J. Detecting gene-gene interactions that underlie human diseases. Nature Reviews Genetics, 2009, 10(6): 392–404
91. Phillips P C. Epistasis — The essential role of gene interactions in the structure and evolution of genetic systems. Nature Reviews Genetics, 2008, 9(11): 855–867
92. Purcell S, Neale B, Todd-Brown K, Thomas L, Ferreira M A, Bender D, Maller J, Sklar P, de Bakker P I, Daly M J, Sham P C. PLINK: A tool set for whole-genome association and population-based linkage analyses. American Journal of Human Genetics, 2007, 81(3): 559–575
93. Ritchie M D, Hahn L W, Moore J H. Power of multifactor dimensionality reduction for detecting gene-gene interactions in the presence of genotyping error, missing data, phenocopy, and genetic heterogeneity. Genetic Epidemiology, 2003, 24(2): 150–157
94. Xu L, Amari S. Combining classifiers and learning mixture-of-experts. In: Ramón J, Dopico R, Dorado J, Pazos A, eds. Encyclopedia of Artificial Intelligence. Hershey, PA: IGI Global, 2008, 318–326
95. Tu S K, Chen R S, Xu L. A binary matrix factorization algorithm for protein complex prediction. Proteome Science, 2011, 9(Suppl 1): S18

Lei XU is a chair Professor of The Chinese University of Hong Kong (CUHK), Chang Jiang Chair Professor of Peking University, a guest Professor of Institute of Biophysics, Chinese Academy of Sciences, an honorary Professor of Xidian University. He graduated from Harbin Institute of Technology (HIT) by the end of 1981, and completed his master and Ph.D thesis at Tsinghua University during 1982–1986. Then, he joined Department of Mathematics, Peking University in 1987 first as postdoc and then exceptionally promoted to associate professor in 1988 and to a full professor in 1992. During 1989–1993, he was senior researcher, research associate, and postdoc in Finland, Canada, and USA, including Harvard and MIT. He joined CUHK in 1993 as senior lecturer, as professor in 1996, and chair professor in 2002. He has published a number of well-cited papers on neural networks, statistical learning, and pattern recognition, e.g., his papers got over 3900 citations according to SCI and over 7100 citations according to Google Scholar (GS), with the first 10 papers scored over 2310 (SCI) and 4600 (GS). One single paper has scored 857 (SCI) and 1690 (GS). The H-index is 27 (SCI) and 34 (GS).

He served as associate editor for several journals, including Neural Networks (1995–present) and IEEE Transactions on Neural Networks (1994–1998), and as general chair or program committee chair of a number of international conferences. Moreover, Prof. Xu has served on governing board of International Neural Networks Society (INNS) (2001–2003), INNS Award Committee (2002–2003), and Fellow Committee of IEEE Computational Intelligence Society (CIS) (2006, 2008), chair of IEE CIS Computational Finance Technical Committee (2001–2003), a past president of Asian-Pacific Neural Networks Assembly (APNNA) (1995–1996), and APNNA Award Committee (2007–2009). He has also served as an engineering panel member of Hong Kong RGC Research Committee (2001–2006), a selection committee member of Chinese NSFC/HK RGC Joint Research Scheme (2002–2005), external expert for Chinese NSFC Information Science (IS) Panel (2004–2006, 2008), external expert for Chinese NSFC IS Panel for distinguished young scholars (2009–2010), and a nominator for the prestigious Kyoto Prize (2003, 2007, 2011). Prof. Xu has received several Chinese national academic awards (including 1993 National Nature Science Award) and international awards (including 1995 INNS Leadership Award and the 2006 APNNA Outstanding Achievement Award). He has been elected to an IEEE Fellow since 2001, a Fellow of International Association for Pattern Recognition, and a member of European Academy of Sciences since 2002.