Studies in Computational Intelligence(SCI) 63 : Challenges for Computational Intelligence, pp365-406 , Springer-Verlag Berlin Heidelberg 2007

A Trend on Regularization and Model Selection in Statistical Learning: *A Bayesian Ying Yang Learning Perspective*

Lei Xu

Department of Computer Science and Engineering, Chinese University of Hong Kong, Shatin, NT, Hong Kong, P.R. China

Summary. In this chapter, advances on regularization and model selection in statistical learning have been summarized, and a trend has been discussed from a Bayesian Ying Yang learning perspective. After briefly introducing Bayesian Ying-Yang system and best harmony learning, not only its advantages of automatic model selection and of integrating regularization and model selection have been addressed, but also its differences and relations to several existing typical learning methods have been discussed and elaborated. Taking the tasks of Gaussian mixture, local subspaces, local factor analysis as examples, not only detailed model selection criteria are given, but also a general learning procedure is provided, which unifies those automatic model selection featured adaptive algorithms for these tasks. Finally, a trend of studies on model selection (i.e., automatic model selection during parametric learning), has been further elaborated. Moreover, several theoretical issues in a large sample size and a number of challenges in a small sample size have been presented. The contents consist of

Lei Xu: *A Trend on Regularization and Model Selection in Statistical Learning: A Bayesian Ying Yang Learning Perspective*, Studies in Computational Intelligence (SCI) **63**, 365–406 (2007)
www.springerlink.com

Key words: Statistical learning, Model selection, Regularization, Bayesian Ying-Yang system, Best harmony learning, Best matching, Best fitting, AIC, BIC, Automatic model selection, Gaussian mixture, Local factor analysis, theoretical issues, challenges.

1 Best Fitting vs Over-fitting

Statistical learning is usually referred to a process that a learner discovers certain dependence relation underlying a set of samples $\mathcal{X}_N = \{x_t\}_{t=1}^N$. The learner is equipped with a device or model M to accommodate this dependence relation. Such a relation is featured by a specific structure S^o and a specific setting θ^o taken by a set of parameters θ. Keeping the same structure S^o, we can get a family of specific relations $S^o(\theta)$ by varying θ within a given domain Θ that includes θ^o. Provided that every x_t comes from $S^o(\theta^o)$ without noise or disturbance, if we know S^o but do not directly know θ^o, we can get $\theta = \theta^o$ via the principle of searching one $\theta \in \Theta$ such that $S^o(\theta)$ best fits the samples in $\mathcal{X}_N$, as long as N is large enough. Usually, samples come from $S^o(\theta^o)$ subject to certain uncertainties, e.g., noises, disturbances, random sampling, etc. When N is large enough, we may still get a unique estimate value θ^* to approximate θ^o via this best fitting principle. Such a task of determining θ is usually called *parameter learning*.

The task becomes not so simple in the cases that either S^o is unknown or N is not large enough even when S^o is known. If we do not know S^o, we have to assign an appropriate structure to M. More specifically, a structure is featured by its structure type and its complexity or scale. E.g., considering relations described by $y(x) = a_3x^3 + a_2x^2 + a_1x + a_0$, its structure type is a polynomial and its scale is simply an integer that is equal to 3. For two structures S_a and S_b of a same type, S_a is actually a sub-structure (or S_a is included in S_b, shortly denoted by $S_a \prec S_b$) if S_a has a scale smaller than that of S_b. E.g., a polynomial of the order 2 is a sub-structure in a polynomial of the order 3. For two structures S_a and S_b of different types, if one is not a sub-structure of the other, we can always enlarge the scale of one structure to a large enough one such that it includes the other as a sub-structure. For this reason, we let M to consider a family of structures $S(\theta_{\mathbf{k}}, \mathbf{k})$, where S may not be same as the unknown one of S^o, but is pre-specified by one of typical structures, depending on a specific learning task encountered. Readers are referred to [41] for a number of typical structure types. $\mathbf{k}$ is a tuple that consists of one or

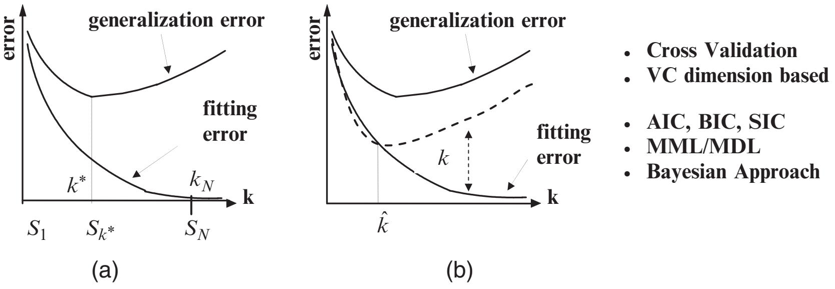


Fig. 1. Fitting error vs generalization error

several integers. By enumerating in a certain manner the values that $\mathbf{k}$ takes, we can get a series of embedded structures $S_1 \prec S_2 \prec \cdots \prec S_k \cdots$ such that $S_{k^*-1} \prec S^o \prec S_{k^*}$.

It is not difficult to find k^* if $\mathcal{X}_N$ comes from $S^o(\theta^o)$ without noise or disturbance. Searching a best value of θ_1 such that $S_1(\theta_1)$ best fits the samples in $\mathcal{X}_N$, there will be a big fitting error. This fitting error will monotonically decrease as k increases. Also, it reaches zero when $k = k^*$ and remains to be zero as k further increases. That is, k^* can be found by the smallest k where a zero fitting error is reached. However, the best fitting error by $S_{k^*}(\theta_{k^*})$ will still be nonzero, if the samples in $\mathcal{X}_N$ have been infected by noises while N is a finite size. As shown in Fig. 1(a), the fitting error will keep to decrease monotonically as k further increases, until it reaches zero at k_N that relates to N and but is usually much larger than k^*. In other words, a large part of structure with a much larger scale has actually been used to fit noises or disturbances. As a result, we can not get k^* by the principle of finding the smallest scale at which the best-fitting error is zero. This is usually called *over-fitting* problem.

We also encounter the same problem even when we known that the samples $\mathcal{X}_N = \{x_t\}_{t=1}^N$ come from $S^o(\theta^o)$ without noise but the size N is not large enough. In such a case, we are unable to determine a unique θ^* via the best fitting principle, because there will be infinite many choices of θ by which the best fitting error is zero. In other words, the samples in $\mathcal{X}_N$ actually come from a unknown sub-structure inside S^o. That is, we are lead to the same situation as the above ones with S^o unknown.

2 Trends on Regularization and Model Selection

2.1 Regularization

In the literatures of statistics, neural networks, and machine learning, many efforts in two directions have been made on tackling the *over-fitting* problem in the past 30 or 40 years. One direction consists of those made under the name of *regularization* that is imposed during parameter learning [35, 24].

Though we do not know the original structure underlying the samples in $\mathcal{X}_N$, we may consider a structure $S_k(\theta_k)$ with its scale large enough to include the original structure, which is also equivalent to the cases that the samples $\mathcal{X}_N = \{x_t\}_{t=1}^N$ come from a known structure $S^o(\theta^o)$ but with N being not large enough. Instead of searching an appropriate substructure, we impose certain constraint on θ or certain regularity on the structure $S(\theta) = S_k(\theta_k)$ with a scale k such that we can find a unique θ^* by best fitting to get a specific structure $S(\theta^*)$ but still with a scale k as an effective approximation to a substructure in a lower scale.

The existing typical regularization techniques can be roughly classified into two types. One is featured by a corrected best fitting criterion in a format of *best-fitting plus correction*, as summarized below:

- *Tikhonov regularization* One popular approach relates to the well known Tikhonov regularization [24, 11, 35], featured by the following format
$$\theta^* = arg \min_{\theta} [F(S(\theta), \mathcal{X}_N) + \lambda P(S(x, \theta))], \tag{1}$$
where $F(S(\theta), \mathcal{X}_N)$ denotes the fitting error for implementing the best fitting principle, and $P(S(x, \theta))$ is usually called a stabilizer that describes the irregularity or non-smoothness of the manifold $S(x, \theta)$ specified by the structure $S(\theta)$. Moreover, λ is called a regularization strength that controls how strong the stabilizer is in action.
- *Ridge regression* It has been widely used in the literature of neural networks (see a summary by Sec. 2.3 in [67]), featured by the following format
$$\theta^* = arg \min_{\theta} [F(S(\theta), \mathcal{X}_N) + \lambda \Omega(\theta)], \tag{2}$$
where $\Omega(\theta)$ denotes a regularized term that attempts to shrink the dynamic range that θ varies. One typical example is $\Omega(\theta) = \|\theta\|^2$.
- *Bayesian approach* Another widely studied apporach is called maximum a posteriori probability (MAP), featured by maximizing the posteriori probability
$$p(\theta|\mathcal{X}_N) = p(\mathcal{X}_N|\theta)p(\theta)/p(\mathcal{X}_N), \tag{3}$$
or equivalently minimizing $-[\ln p(\mathcal{X}_N|\theta) + \ln p(\theta)]$ with $\ln p(\mathcal{X}_N|\theta)$ taking the role of $F(S(\theta), \mathcal{X}_N)$, while $\ln p(\theta)$ takes the role of $\Pi(S(x, \theta))$ and $\Omega(\theta)$.

This type has both one crucial weakness and one key difficulty. The crucial weakness comes from the choices of $P(S(x, \theta))$ in eq.(1), $\Omega(\theta)$ in eq.(2), and $p(\theta)$ in eq.(3), which usually have to be imposed in an isotropic manner.

Thus, regularization works only to a single mode data set in a symmetrical, isotropic, or uniform structure. However, such a situation is usually not encountered for two main reasons. First, a data set may include multiple disconnected substructures. Second, even for a single mode data set in a symmetrical, isotropic, or uniform structure, we need to use a structure with an appropriate scale k^* to fit it. If a structure with a larger sale $k > k^*$ is used, it is desired that the part corresponding to those extra scales can be ignored or discarded in fitting. For an example, given a set of samples (x_t, y_t) that come from a curve $y = x^2 + 3x + 2$, if we use a polynomial of an order $k > 2$ to fit the data set, i.e. $y = \sum_{i=0}^{k} a_i x^i$, we desire to force all the parameters $\{a_i, i \geq 3\}$ to be zero. In other words, we have to treat the parameters $\{a_i, i \geq 3\}$ differently from the parameters $\{a_i, i \leq 2\}$, instead of being in an isotropic or uniform way.

In addition to the above problem, we also encounter a key difficulty on how to appropriately control the strength λ of regularization, which is usually able to be roughly estimated only for a rather simple structure via either handling the integral of marginal density or in help of cross validation, but with very extensive computing costs [31, 32, 33].

The second type of regularization techniques consists of those not directly guided by a corrected best fitting criterion, but rather heuristically based. While some are quite specific problem dependent, some are generally applicable to many problems, for examples we can

- add noises to the samples in $\mathcal{X}_N$;
- add noises to a solution θ^* obtained by certain approach;
- terminate a learning process before it converges.

Though the adding noise approaches can be qualitatively related to the Tikhonov regularization [5], we do not know what type of noises to add and thus usually add a Gaussian noise with a variance λ, which is still an isotropic type regularization. In other words, we still can not avoid being suffering from the previously discussed crucial weakness. Also, it is very difficult to determine an appropriate variance λ.

As to the early termination approach, it has also been qualitatively related to Tikhonov regularization in some simple structure. However, it is very difficult to guide when to terminate.

Actually, all the regularization efforts suffer the previously discussed crucial weakness and difficulty, which come from its nature of searching a structure $S(\theta^*)$ with a scale k to approximate a substructure in a lower scale, while the part related to those extra scales has not been discarded but still in action to blur those useful ones. To tackle the problems, we turn to consider the other direction that consists of those efforts made under the name of *model selection*, featured by searching a structure with an appropriate scale k^*.

2.2 Model Selection

As discussed previously in Fig. 1, we can not find an appropriate k^* according to the best fitting principle. Several theories or principles have been proposed to guide the selection of k^*, which can be roughly classified into two categories.

One directly bases on the generalization error, i.e., the fitting error of an estimated $S_k(\theta_k^*)$ not only on the samples in $\mathcal{X}_N$ but also on all the other samples from $S^o(\theta^o)$ subject to noises and certain uncertainty. Using $p_o(x) = p(x|S^o(\theta^o))$ to describe that x comes from $S^o(\theta^o)$ subject to certain noises or uncertainty, and letting $\varepsilon(x,k)$ to denote the fitting error of x by $S_k(\theta_k^*)$, if we measure the following expected error (also called generalization error):

$$E_g(k) = \int \varepsilon(x,k) p_o(x) dx, \tag{4}$$

which takes in consideration not only the best fitting error on the samples in $\mathcal{X}_N$ but also all the other samples from $S^o(\theta^o)$.

However, it is impossible to estimate the generalization error by knowing all the samples from $S^o(\theta^o)$. We have to face the difficulty of estimating the generalization error merely based on the samples in $\mathcal{X}_N$. Two representative theories for this purpose are as follows:

- *Estimating generalization error by experiments* Studies of this type are mostly made under the name of cross-validation (CV) [31, 32, 33, 28], by which generalization error is estimated in help of experiments of making training and testing via repeatedly dividing a same set of samples into a different training set and a different testing set.
- *Estimating bounds of general error via theoretical analysis* Though it is theoretically impossible to get an analytical expression for the generalization error merely based on the samples in $\mathcal{X}_N$, we may estimate some rough bound $\Delta(k)$ on the difference between the best fitting error and the generalization error. As shown in Fig. 1(b), we consider

$$\hat{k} = arg \min_k J(k, \theta_k^{fit}), \ J(k, \theta_k^{fit}) = F(\mathcal{X}_N, \theta_k^{fit}) + \Delta(k), \tag{5}$$

  where $\theta_k^{fit} = arg \min_{\theta_k} F(\mathcal{X}_N, \theta_k)$, and $F(\mathcal{X}_N, \theta_k)$ is a cost measure for implementing a best fitting, e.g., it is usually the negative likelihood function $-\ln p(\mathcal{X}_N|\theta_k)$. One popular example for this $\Delta(k)$ is the VC dimension based learning theory [39]. A bound $\Delta(k)$ relates to not only N and the fitting error $F(\mathcal{X}_N, \theta_k^{fit})$, but also a key index called VC dimension. Qualitatively, such a bound $\Delta(k)$ is useful for theoretical understanding. However, it is usually difficult to implement because the VC dimension is very difficult to estimate except for some simple cases.

The other category summarizes all the other efforts not directly based on estimating the generalization error. They are closely related but proposed from different aspects, as summarized below:

- *Minimizing information divergence* The discrepancy between the true model and the estimated model is minimized via minimizing $KL(p_o \| p_{\theta_k})$, where $p_{\theta_k} = p(x|\theta_k) = p(x|S_k(\theta_k))$ and $KL(p\|q)$ is the well known Kullback-Leiber information, i.e.,

$$KL(p\|q) = \int p(x) \ln \frac{p(x)}{q(x)} dx. \tag{6}$$

It further follows that $KL(p_o\|p_{\theta_k}) = H(p_o\|p_o) - H(p_o\|p_{\theta_k})$, where

$$H(p\|q) = \int p(x)\ln q(x)dx. \tag{7}$$

Its negation $-H(p\|q)$ is also called the cross entropy, and $-H(p\|p)$ is the entropy of $p(x)$. Since $H(p_o\|p_o)$ is constant, minimizing $KL(p_o\|p_{\theta_k})$ is equivalent to maximizing $H(p_o\|p_{\theta_k})$. If the unknown true density $p_o(x)$ is simply replaced by the empirical density from the sample set $\mathcal{X}_N$, i.e.,

$$p_{\mathcal{X}_N}(x) = \frac{1}{N}\sum\nolimits_{t=1}^{N}\delta(x - x_t), \tag{8}$$

maximizing $H(p_{\mathcal{X}_N}\|p_{\theta_k})$ becomes the maximum likelihood function, which gives $\theta_k^{ML} = arg\max_{\theta_k} H(p_{\mathcal{X}_N}\|p_{\theta_k})$. However, as discussed before, the best fitting measure $H(p_{\mathcal{X}_N}\|p_{\theta_k^{ML}})$ will monotonically decrease as k and thus is not good to be used for selecting k. A better choice is to use the unknown $E_{\mathcal{X}_N}[H(p_{\mathcal{X}_N}\|p_{\theta_k})]_{\theta_k=\theta_k^{ML}}$. To estimate it, we consider the bias

$$b(k,N) = E_{\mathcal{X}_N}\{E_{\mathcal{X}_N}[H(p_{\mathcal{X}_N}\|p_{\theta_k})]_{\theta_k=\theta_k^{ML}}\} - E_{\mathcal{X}_N}H(p_o\|p_{\theta_k^{ML}}), \tag{9}$$

which relates to both k and N. With this bias $b(k,N)$ estimated, we are lead to eq.(5) with

$$F(\mathcal{X}_N, \theta_k^{fit}) = H(p_{\mathcal{X}_N}\|p_{\theta_k^{ML}}),\ \mathit{\Delta}(k) = b(k,N) \tag{10}$$

for selecting k. Along this line, we are further lead to the well known Akaike information criterion (AIC) with $b(k,N) = 0.5d_k/N$ [1, 2, 3] and a number of its extensions AICB, CAIC, etc, [34, 6, 7, 14, 8].

- *Optimizing marginal likelihood* Introducing a prior $p(\theta_k)$, we consider to maximize the likelihood of the following marginal distribution:

$$p(x|S_k) = \int p(x|S_k(\theta_k))p(\theta_k)d\theta_k. \tag{11}$$

Instead of solving this integration, we consider $h(\theta_k) = \ln p(x|S_k(\theta_k))$ that is expanded into a second order Taylor series with respect to θ_k around θ_k^{ML}. Let $p(\theta_k) = 1$ noninformatively inside the intergal, we get

$$\begin{aligned} p(x|S_k) &= p(x|S_k(\theta_k^{ML}))\int p(\theta_k)e^{-0.5N(\theta_k-\theta_k^{ML})^T I(\theta_k^{ML})(\theta_k-\theta_k^{ML})}d\theta_k, \\ &= p(x|S_k(\theta_k^{ML}))(2\pi)^{0.5d_k}|I(\theta_k^{ML})N|^{-0.5} \end{aligned} \tag{12}$$

where d_k is the dimension of θ_k, and $I(\theta_k)$ is the observed Fisher information matrix. After ignoring some terms approximately, we are lead to eq.(5) again with

$$F(\mathcal{X}_N, \theta_k^{fit}) = H(p_{\mathcal{X}_N}\|p_{\theta_k^{ML}}),\ \mathit{\Delta}(k,N) = 0.5\frac{d_k \ln N}{N}, \tag{13}$$

which is usually referred as Bayesian inference criterion (BIC) [29, 15, 23].

- *Ockham Razor (minimizing two parts of coding)* The idea is to minimize the sum of the description length for the model and the description length for residuals that the model fails to fit. Typical examples include those studies under the name of Minimum Message Length (MML) theory [36, 37, 38] and the name of Minimum Description Length (MDL) [26, 27, 22, 9, 13]. Though being different from the above BIC type approach conceptually, the implementation of either MML or MDL actually crash back to be exactly equivalent to the above BIC type approach.

Though each of the above approaches can provide a criterion $J(k, \theta_k^*)$ in a format of eq.(5), we still have to face two problems for selecting an appropriate scale k^*. First, in the cases that the size N of samples is small or not large enough, each criterion actually provides a rough estimate that can not guarantee to give k^*, and even results in a wrong result especially when k consists of several integers to enumerate. Moreover, one criterion works better in this case and the other criterion may work better in that case, none can be said to be better than the others. Second, in addition to the performance problem, another difficulty is its feasibility in implementation, because it has to be made in the following two phases:

Phase 1: Enumerate k within a range from k_d to k_u, that is assumed to contain the optimal k^*. For each specific k, we make *parameter learning* to get $\theta_k^{fit} = arg\min_{\theta_k} F(\mathcal{X}_N, \theta_k)$ according to the best fitting principle (e.g., minimizing a square error or maximizing a likelihood function).

Phase 2: Select $k^* = arg\min_k J(k, \theta_k^{fit})$ for every k within $[k_d, k_u]$.

This two-stage implementation is very expensive in computing, which makes it infeasible in many real applications.

2.3 Model Selection: from Incremental to Automatic

There are also efforts made in literatures towards the difficulty of the above two-stage implementation. One type of efforts is featured by an incremental implementation. Parameter learning is made incrementally in a sense that it attempts to incorporate as much as possible what learned at k into the learning procedure at $k+1$. Also, the calculation of $J(k, \theta_k^*)$ is made incrementally. Such an incremental implementation can indeed save computing costs in certain extent. However, parameter learning has to be made still by enumerating the values of k, and computing costs are still very high. As k increases to $k+1$, an incremental implementation of parameter learning may also lead to suboptimal performance because not only those newly added parameters but also the old parameter set θ_k have to be re-learned.

Another type of efforts has been made on a widely encountered category of structures, with each consisting or composing of k individual substructures, e.g., a Gaussian mixture structure that consists of k Gaussian components. A local error criterion is used to check whether a new sample x belongs to each substructure. If x is regarded as not belonging to any of the k substructures, the $k+1$-th substructure is added to accommodate this new x. This incremental implementation for determining k is much faster. However, the local evaluating nature makes it very easy to be trapped into a poor performance, except for some special cases that $\mathcal{X}_N = \{x_t\}_{t=1}^N$ come from substructures that are well separated from each other.

Another new road has also been explored for more than ten years, with a feature that model selection can be implemented automatically during parameter learning, in a sense that making parameter learning on a structure $S_k(\theta_k)$ with its scale large enough to include the correct structure, will not only determine parameters but also automatically shrink its scale to an appropriate one, while those extra substructures are discarded during learning. It combines the good feature of *regularization* and the good feature of *model selection.* On one hand, it takes the good feature of *regularization* (i.e., parameter learning is only implemented on a structure $S_k(\theta_k)$ with a larger scale), but discards the crucial problem of regularization (i.e., those extra substructures are still kept to blur the useful ones). On the other hand, it takes the good feature of *model selection*, i.e., only a structure with an appropriate scale is in action without any extra substructures to deteriorate its performance. Moreover, it not only keeps the model selection performance as good as that by a two-stage implementation, but only performs parameter learning only once with a drastic reduction in computing costs.

One early effort along such a new road started from Rival Penalized Competitive Learning (RPCL) for clustering analysis and detecting curves in an image [64, 66]. A structure in a scale k consists of k individual substructures, with each being simply one point as one cluster's center. Initially, k is given a value larger than the appropriate number of clusters. A coming sample x is allocated to one of the k centers via competition, and the winning center moves a little bit to adapt the information carried by this sample. Moreover, the rival (i.e., the second winner) is repelled a little bit away from the sample to reduce a duplicated information allocation. Driving those extra centers away, this rival penalized mechanism will keep an appropriate number of centers. In other words, RPCL makes the number of clusters determined automatically during learning. This is a favorable feature that the conventional competitive learning or clustering algorithm (e.g., k-means) does not have. RPCL has been further extended from spheral clusters to elliptic clusters via Gaussian mixture [55, 52, 49]. Readers are referred to [40, 47] for a recent elaboration and to [17, 18, 16] for successful applications.

RPCL learning was heuristically proposed on a bottom level (i.e., a level of learning dynamics or updating rule), which is quite different from our previous discussions on a global level of using one learning principle or theory to guide parameter learning and model selection in a top-down manner. Proposed firstly in [59] and systematically developed in past years [53, 51, 52, 49, 47, 42, 43, 41], the Bayesian Ying-Yang (BYY) harmony learning is such a global level theory that guides various statistical learning tasks with model selection achieved automatically during parameter learning.

3 BYY Harmony Learning: A Trend on Regularization and Model Selection

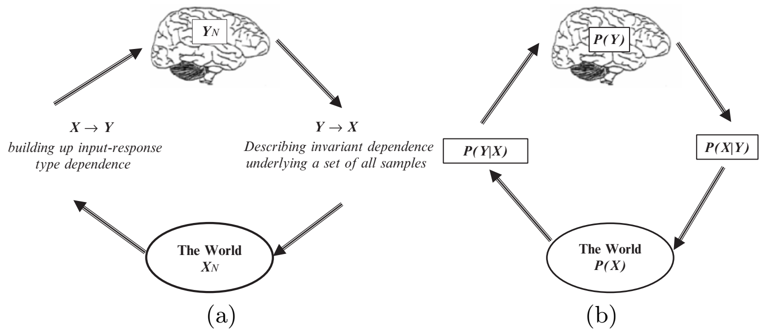
3.1 Bayesian Ying-Yang System

Instead of letting M to consider a parametric structure for directly best fitting the observed samples $\mathcal{X}_N$, M is considered or designed as a system that jointly describes the observed samples and their inner representations $\mathcal{Y}$ via two different but complementary parts. As shown in Fig. 2(a), one is named as *Yang* that consists of one component for representing $\mathcal{X}_N$ and one component for describing a path from $\mathcal{X}_N$ to $\mathcal{Y}$. The other part is named as *Ying* that consists of one component for representing $\mathcal{Y}$ and one component for describing a path from $\mathcal{Y}$ to $\mathcal{X}_N$. Each of the four components may have several choices for its corresponding structure. The four components can be integrated into a system in more than one choices under the name of architecture. In such a Ying-Yang system, a principle of using a structure in a specific type to best fit $\mathcal{X}$ can be generalized into a principle for a Ying-Yang system to best match each other, which includes using a structure to directly best fit $\mathcal{X}$ as a subtask.

The Ying-Yang system can be further formulated in a statistical framework by considering the joint distribution of $\mathcal{X}$ and $\mathcal{Y}$, which can be described via the following two types of Bayesian decomposition:

$$p(\mathcal{X}, \mathcal{Y}) = p(\mathcal{Y}|\mathcal{X})p(\mathcal{X}),\ q(\mathcal{X}, \mathcal{Y}) = q(\mathcal{X}|\mathcal{Y})q(\mathcal{Y}). \tag{14}$$

In a compliment to the famous Chinese ancient Ying-Yang philosophy, the decomposition of $p(\mathcal{X}, \mathcal{Y})$ coincides the Yang concept with $p(\mathcal{X})$ describing samples from an observable domain (called the Yang space) and $p(\mathcal{Y}|\mathcal{X})$ describing the forward path from $\mathcal{X}_N$ to $\mathcal{Y}$ (called the Yang pathway). Thus, $p(\mathcal{X}, \mathcal{Y})$ is called the Yang machine. Similarly, $q(\mathcal{X}, \mathcal{Y})$ is called the Ying machine with $q(\mathcal{Y})$ describing representations in an invisible domain (thus regarded as a Ying space) and $q(\mathcal{X}|\mathcal{Y})$ describing the backward path (called the


Y_N
X → Y
building up input-response type dependence
Y → X
Describing invariant dependence underlying a set of all samples
The World
X_N
P(Y)
P(Y|X)
P(X|Y)
The World
P(X)

(a) (b)

Fig. 2. (a) Ying-Yang system, (b) Bayesian Ying-Yang system

Ying pathway). As shown in Fig. 2(b), such a pair of Ying-Yang machines is called *Bayesian Ying-Yang (BYY) system.*

Each of the above four components may have more than one choices for its structure, as summarized below:

- $p(\mathcal{X})$ is either a nonparametric estimation via data smoothing, e.g.,
$$p_h(\mathcal{X}_N) = \prod_{t=1}^{N} G(x_t|\bar{x}_t, h^2 I), \tag{15}$$
or a direct use of the samples in $\mathcal{X}_N$, i.e., by $p_0(\mathcal{X}_N) = p_h(\mathcal{X}_N)|_{h=0}$. That is, each specific sample $\bar{x}_t$ is blurred or smoothed by a Gaussian noise with a variance h^2, resulting in a Gaussian random variable x_t.
- The structure of $q(\mathcal{Y})$ takes at least three crucial roles. One is bridging the Ying and Yang two parts. The other is specifying the nature of learning tasks via a number of choices summarized by a general structure [41]. The third role is that the scale of its structure actually dominates the scale of the system architecture, as to be introduced in the next subsection.
- With the settlements of $q(\mathcal{Y})$ and $p(\mathcal{X})$, each of $p(\mathcal{Y}|\mathcal{X})$ and $q(\mathcal{X}|\mathcal{Y})$ has also more than one different choices.

Moreover, there are different ways for integrating the four components together, which result in different system architectures.

As a result, $p(\mathcal{X}, \mathcal{Y})$ and $q(\mathcal{X}, \mathcal{Y})$ in eq.(14) are actually not the same though both represent the same joint distribution of $\mathcal{X}$ and $\mathcal{Y}$. In such a formulation, the best fitting principle is generalized into a principle that $p(\mathcal{X}, \mathcal{Y})$ and $q(\mathcal{X}, \mathcal{Y})$ best match each other. This match can be measured by the Kullback divergence [59] and several non-Kullback divergences [57, 45], summarized in the following general expression:

$$D(p\|q, \theta_k) = \int p(\mathcal{Y}|\mathcal{X})p(\mathcal{X})f(\frac{p(\mathcal{Y}|\mathcal{X})p(\mathcal{X})}{q(\mathcal{X}|\mathcal{Y})q(\mathcal{Y})})d\mathcal{X}d\mathcal{Y}, \tag{16}$$

where $f(r)$ is a convex function. Particularly, we have the Kullback divergence when $f(r) = \ln r$. In this setting, our task becomes to specify each of the four components via determining all the unknowns subject to all the known parts, i.e., $\mathcal{X}_N$ and the pre-specified structure of each component. In a summary, we have

$$\min_{\substack{p(\mathcal{X}),\ p(\mathcal{Y}|\mathcal{X}),\ q(\mathcal{X}|\mathcal{Y}), q(\mathcal{Y}), \textit{and } \theta_k \textit{ subject to} \\ \textit{their pre-specified structures and } \mathcal{X}_N}} D(p\|q, \theta_k). \tag{17}$$

In the special case that $p(\mathcal{Y}|\mathcal{X})$ is free of any pre-structure, minimizing $D(p\|q, \theta_k)$ with respect to $p(\mathcal{Y}|\mathcal{X})$ will lead to a special case that is equivalent to using $q(\mathcal{X}) = \int q(\mathcal{X}|\mathcal{Y})q(\mathcal{Y})d\mathcal{Y}$ to best fit $\mathcal{X}_N$ in a sense of the maximum likelihood principle. This nature, together with the feature that the special settings of $q(\mathcal{Y})$ as well as of other three components $q(\mathcal{X}|\mathcal{Y})$, $p(\mathcal{Y}|\mathcal{X})$ and $p(\mathcal{X})$ lead to specific structures of a number of existing typical learning tasks, makes a number of existing statistical learning approaches summarized in a unified perspective with new variants and extensions [44, 45, 41].

3.2 BYY Harmony Learning

Still, we encounter an over-fitting problem for such a best Ying-Yang match by eq.(17). Similar to what discussed in Sec. 1, we consider the BYY system with a family of system architectures of the same type but in different scales.

A system architecture type is an integration of the four components with each in its own specific structure type. As introduced in the previous subsection, the structure type of $q(\mathcal{Y})$ takes a leading role, and the scale of the entire system architecture is dominated by the scale of the structure for $q(\mathcal{Y})$. We can denote this scale by an integer k that usually represents an enumeration of a set of integers $\mathbf{k}$ embedded within the structure of $q(\mathcal{Y})$. Also we can use θ_k to denote all the unknowns in an architecture of a scale k. Suffering an expensive computing cost, we may learn a best value $\theta^*_{k^*}$ at a best scale k^* via the following two-phase implementation:

Phase 1: Enumerate a series of architectures of a same type but in different scales. For each one at k, we make *parameter learning* to get $\theta^*_k = arg\min_{\theta_k} D(p\|q,\theta_k)$.

Phase 2: Select $k^* = arg\min_k J(k,\theta^*_k)$ according to one of the existing model selection criteria.

Much more importantly, the Ying-Yang system is motivated together with the ancient Chinese philosophy that the Ying and the Yang should be best harmony in a sense that two parts should not only best match but also are matched in a compact way. Applying this philosophy into a BYY system, we have a best harmony principle in the following twofold sense:

- *Best matching* the difference between the two Bayesian representations in eq.(14) should be minimized.
- *Least complexity* the resulted architecture for the BYY system should be in a least complexity, i.e., its inner representation has a least complexity.

The above can be further mathematically measured by the following functional

$$H(p\|q,\theta_k) = \int p(\mathcal{Y}|\mathcal{X})p(\mathcal{X})f(q(\mathcal{X}|\mathcal{Y})q(\mathcal{Y}))\mu(d\mathcal{X})\mu(d\mathcal{Y}) - \ln Z, \qquad (18)$$

where $f(r)$ is again a convex function as in eq.(16), and a most useful case is $f(r) = \ln r$. Instead of eq.(17), we specify the four components via determining all the unknowns subject to all the known parts as follows [59, 51, 49]:

$$\max_{\substack{p(\mathcal{X}),\ p(\mathcal{Y}|\mathcal{X}),\ q(\mathcal{X}|\mathcal{Y}), q(\mathcal{Y}),\ \textit{and}\ \theta_k\ \textit{subject to}\\ \textit{their pre-specified structures and}\ \mathcal{X}_N}} H(p\|q,\theta_k), \qquad (19)$$

which guides not only learning on θ^*_k but also model selection on k^*. This is a significant difference from the conventional approaches, by which θ^*_k is learned under a best fitting principle but k^* is selected in help of another learning theory.

The term $\ln Z$ imposes certain regularization into learning on $\mathcal{X}_N$ with a small size N, which will be discussed in Sec. 3.5. Here, we give a further insight on $H(p\|q)$ via the following decomposition

$$\begin{array}{l} H(p\|q,\theta_k) = H_{x|y} + H_y, \\ H_{x|y} = \int p(\mathcal{Y}|\mathcal{X})p(\mathcal{X}) \ln q(\mathcal{X}|\mathcal{Y})\mu(d\mathcal{X})\mu(d\mathcal{Y}), \\ H_y = \int p(\mathcal{Y}) \ln q(\mathcal{Y})\mu(d\mathcal{Y}), \ p(\mathcal{Y}) = \int p(\mathcal{Y}|\mathcal{X})p(\mathcal{X})\mu(d\mathcal{X}). \end{array} \tag{20}$$

On one hand, the term $H_{x|y}$ accounts for a best fitting of the samples in $\mathcal{X}_N$ by $q(\mathcal{X}|\mathcal{Y})$ in help of the corresponding inner representation $\mathcal{Y}$. If $p(\mathcal{X})$ is given by eq.(8) and a set $\mathcal{Y}_N$ is given for pairing $\mathcal{X}_N$, $H_{x|y}$ degenerates into the likelihood function of $q(\mathcal{X}|\mathcal{Y})$. On the other hand, the term H_y accounts for two purposes. When the structure of $q(\mathcal{Y})$ is not pre-imposed with too much constraint, maximizing H_y results in $q(\mathcal{Y}) = p(\mathcal{Y})$ such that $-H_y$ becomes exactly the entropy of the inner representation. Thus, maximizing H_y leads to an inner representation in a least complexity. Usually, $q(\mathcal{Y})$ is pre-imposed with different structures for different learning tasks, maximizing H_y forces the resulted $p(\mathcal{Y})$ to satisfy these constraints correspondingly. It can be observed that $H_{x|y}$ increases while H_y decreases as the scale k increases, which trades off for an appropriate k^*. In other words, $H(p\|q,\theta_k)$ can be used at Phase 2 of a two-phase implementation given at the beginning of this subsection, in place of a model selection criterion. That is, we get $\theta_k^* = arg\min_{\theta_k} D(p\|q,\theta_k)$ at Phase 1, and then, as shown in Fig. 3(a), we select a best k^* at Phase 2 by

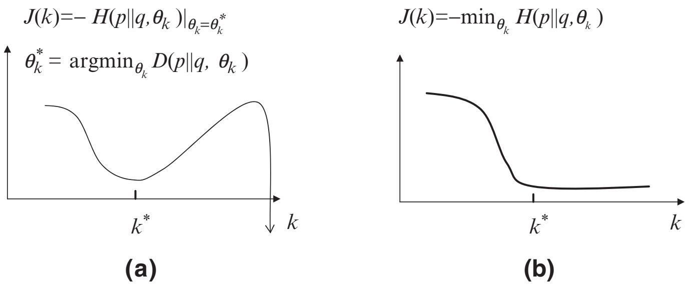
$$k^* = arg\min_k \ J(k), \ J(k) = -H(p\|q,\theta_k)|_{\theta_k=\theta_k^*}. \tag{21}$$

With this two-phase implementation as a link, we can compare the performances of this new criterion with those typical model selection criteria discussed in Sec. 2.2.

3.3 BYY Harmony Learning and Automatic Model Selection

The BYY harmony learning by eq.(19) has a salient advantage that an appropriate scale k^* can be obtained by implementing parameter learning only once on an architecture in a larger scale.

Considering $q(\mathcal{Y})$ in a scale reducible structure that its scale k can be effectively reduced into a smaller one by forcing a critical subset of parameters within the structure becoming zero. That is, we consider a distribution $q(y|\theta_k^y), \theta_k^y \in \Theta_k^y$ that demonstrates its maximum scale k^y when θ_k^y takes values within some specific domain $\hat{\Theta_k}^y$ of Θ_k^y while it effectively reduces into a smaller scale when a critical subset ϕ_k of θ_k^y becomes zero. For an example, a mixture distribution $q(y|\theta_k^y) = \sum_{\ell=1}^k \alpha_\ell q(y|\phi_\ell)$ has a scale k when $\alpha_\ell > 0$ for all ℓ, but will reduce into a scale $k-1$ if one $\alpha_\ell = 0$. For another example, an independent product $q(y|\theta_k^y) = \prod_{j=1}^m q(y^{(j)}|\theta_k^{y^{(j)}})$ has a scale m in general



(a) (b)

Fig. 3. Selection of an appropriate k^*

but a reduced scale $m-1$ when there is one j with $var(y^{(j)})=0$, i.e., the variance parameter of $y^{(j)}$ becomes zero. Readers are referred to Sec. 22.5 in [44], Sec. 23.3.2 in [45], Sec. II(B) in [42], and Sec. III(C) in [41].

For $q(\mathcal{Y})$ in a scale reducible structure, we have two different types of choices. First, let

$$J(k) = -\max_{\theta_k,\ subject\ to\ \theta_k^y \in \hat{\Theta}_k{}^y} H(p\|q,\theta_k^*), \tag{22}$$

we are lead to a case as shown in Fig. 3(a) for a two-phase implementation. E.g., we get such a case when $\alpha_\ell = 1/k$ for all ℓ or $var(y^{(j)})=1$ for all j. Second, we let $J(k) = -\max_{\theta_k} H(p\|q,\theta_k)$ without any constraint, maximizing H_y will push the part $\theta_k^{y\ (2)}$ of those extra substructures to zeros such that $q(\mathcal{Y})$ effective shrinks to a scale smaller than k. As a result, the curve shown in Fig. 3(a) becomes the curve shown in Fig. 3(b).

In other words, considering $q(\mathcal{Y})$ in a scale reducible structure with an initial scale k that is larger than an appropriate one, we can implement the following parameter learning

$$\max_{\theta_k} H(p\|q,\theta_k), \tag{23}$$

which results in not only a best θ_k^* but also an appropriate k^* determined automatically during this learning. Readers are referred to [41, 43, 44, 49].

3.4 BYY Model Selection Criteria on a Small Size of Samples

In a situation that the sample size N is too small, the performance of automatic model selection by the BYY harmony learning will deteriorate, and we have to turn back to a two phase implementation. For this purpose, we seek to find improved model selection criteria from eq.(21).

We consider a more general BYY system with an augmented inner-system representation $\mathcal{R}$ that consists of not only $\mathcal{Y}$ featured by a per sample pairing relation between $\mathcal{X}_N$ and $\mathcal{Y}_N$ (i.e., each x_t gets its inner representation y_t), but also all the unknown parameters θ_k (including h in eq.(15)). With such an extension, eq.(14) becomes

$$p(\mathcal{X},\mathcal{R}) = p(\mathcal{R}|\mathcal{X})p(\mathcal{X}),\ q(\mathcal{X},\mathcal{R}) = q(\mathcal{X}|\mathcal{R})q(\mathcal{R}). \tag{24}$$

Specifically, $q(\mathcal{R}) = q(\mathcal{Y},\theta_k) = q(\mathcal{Y}|\theta_k)q(\theta_k)$ that consists of two parts. One is $q(\theta_k)$ that describes a priori distribution for the values that θ_k may take. The other is actually the previous one under the notation $q(\mathcal{Y})$, which is featured by a family of parametric functions that vary as a set of parameters θ_k^y that is a subset of θ_k. That is, $q(\mathcal{Y}) = q(\mathcal{Y}|\theta_k) = q(\mathcal{Y}|\theta_k^y)$. Coupling with the representation of $\mathcal{Y}$, $q(\mathcal{X}|\mathcal{R}) = q(\mathcal{X}|\mathcal{Y},\theta_k) = q(\mathcal{X}|\mathcal{Y},\theta_k^{xy})$ is actually the previous one under the notation $q(\mathcal{X}|\mathcal{Y})$, defining a family of parametric functions with a set of parameters θ_k^{xy} that is also a subset of θ_k. Moreover, $p(\mathcal{R}|\mathcal{X}) = p(\mathcal{Y},\theta_k|\mathcal{X}) = p(\mathcal{Y}|\mathcal{X},\theta_k)p(\theta_k|\mathcal{X})$ that comprises of two parts. $p(\mathcal{Y}|\mathcal{X},\theta_k^{yx})$ is actually the previous one under the notation $p(\mathcal{Y}|\mathcal{X})$, associated with a set of parameters θ_k^{yx} that is another subset of θ_k too. The other part $p(\theta_k|\mathcal{X})$ describes the uncertainty of estimating θ_k from $\mathcal{X}$, which is actually the posteriori counterpart of the a priori $q(\theta_k)$.

Correspondingly, we can observe that the harmony functional by eq.(18) actually comes from

$$H(p\|q) = \int p(\mathcal{R}|\mathcal{X})p(\mathcal{X})f(q(\mathcal{X}|\mathcal{R})q(\mathcal{R}))\mu(d\mathcal{X})\mu(d\mathcal{R}). \tag{25}$$

In the case $f(r) = \ln r$, it can be further rewritten into

$$\begin{aligned}
&H(p\|q) = \int p(\theta_k|\mathcal{X})H(p\|q,\theta_k)d\theta_k \\
&H(p\|q,\theta_k) = \int p(\mathcal{Y}|\mathcal{X},\theta_k^{yx})p_h(\mathcal{X})\ln[q(\mathcal{X}|\mathcal{Y},\theta_k^{xy})q(\mathcal{Y}|\theta_k^y)]d\mathcal{X}d\mathcal{Y} - Z(\theta_k), \\
&Z(\theta_k) = -\ln q(\theta_k),
\end{aligned} \tag{26}$$

where $H(p\|q,\theta_k)$ is actually the one given in eq.(18) at the case $f(r) = \ln r$.

Given the structures of $q(\mathcal{Y}|\theta_k^y)$, $q(\mathcal{X}|\mathcal{Y},\theta_k^{xy})$, and $p(\mathcal{Y}|\mathcal{X},\theta_k^{yx})$, the task of learning is featured by $\max_{\{p(\theta_k|\mathcal{X}),\ \mathbf{k}\}} H(p\|q)$. By expanding $H(p\|q,\theta_k)$ with respect to θ_k around the following θ_k^* up to the second order and ignoring its first order term since $\nabla_{\theta_k}H(p\|q,\theta_k) = 0$ at $\theta_k = \theta_k^*$, the task can be approximately decomposed into the following two parts:

$$\begin{aligned}
&\theta_k^* = arg\max_{\theta_k} H(p\|q,\theta_k),\ \mathbf{k}^* = arg\min_{\mathbf{k}} J(\mathbf{k}),\ J(\mathbf{k}) = -H(p\|q), \\
&H(p\|q) = H(p\|q,\theta_k^*) - 0.5d(\theta_k^*), \\
&d(\theta_k) = -Tr[\Sigma_{\theta_k}\frac{\partial^2 H(p\|q,\theta_k)}{\partial\theta_k\partial{\theta_k}^T}]_{\theta_k=\theta_k^*}, \\
&\Sigma_{\theta_k} = \int(\theta_k-\theta_k^*)(\theta_k-\theta_k^*)^T p(\theta_k|\mathcal{X})d\theta_k.
\end{aligned} \tag{27}$$

That is, to get rid of the difficulty of estimating $p(\theta_k|\mathcal{X})$ and the related computing cost, we can implement learning in two phases as follows:

Phase 1: Enumerate $\mathbf{k}$ for a series of architectures of a same type but in different scales. For each candidate, we estimate $\theta_k^* = arg \max_{\theta_k} \ H(p\|q, \theta_k)$.

Phase 2: Select a best architecture by $\mathbf{k}^* = arg \min_{\mathbf{k}} J(\mathbf{k})$, where $d(\theta_k^*)$ can be further approximately simplified into an integer as follows:

$$d(\theta_k) = \begin{cases} d_k, & (a) an\ under-constraint\ choice, \\ 2d_k, & (b) an\ over-constraint\ choice, \end{cases} \tag{28}$$

where d_k is the number of free parameters in θ_k.

Eq. (28) comes from a reason related to the celebrated Cramer-Rao inequality. We roughly regard that the obtained θ_k^* suffers a uncertainty under $p(\theta_k|\mathcal{X})$ with a covariance matrix Σ_{θ_k} such that $\Sigma_{\theta_k} \frac{\partial^2 H(p\|q,\theta_k)}{\partial \theta_k \partial {\theta_k}^T} \approx I$ at $\theta_k = \theta_k^*$, especially when we consider a noninformative priori $q(\theta_k) = 1$ or $\ln q(\theta_k) = 0$, which leads to eq.(28)(a). Generally, it may be too crude to simply count the number of parameters in θ_k. Instead, $d(\theta_k^*)$ is an effective number closely related to how $p(\theta_k|\bar{\mathcal{X}}_N)$ is estimated. For an estimator $\theta_k^* = T(\bar{\mathcal{X}}_N)$ basing on a sample set $\bar{\mathcal{X}}_N$, if this estimator is unbiased to its true value θ^o, it follows from the celebrated Cramer-Rao inequality that $p(\theta_k|\bar{\mathcal{X}}_N)$ can be asymptomatically regarded as $p(\theta_k|\bar{\mathcal{X}}_N) = G(\theta_k|\theta^o, [NF(\theta^o)]^{-1})$ with $F(\theta) = -\frac{1}{N} \frac{\partial^2 \ln q(\bar{\mathcal{X}}_N|\theta)}{\partial \theta \partial \theta^T}$, and thus we have $\Sigma_{\theta_k} = \int (\theta_k - \theta^o + \theta^o - \hat{\theta}_k)(\theta_k - \theta^o + \theta^o - \hat{\theta}_k)^T G(\theta_k|\theta^o, [NF(\theta^o)]^{-1}) d\theta_k \approx 2[NF(\theta^o)]^{-1}$. Moreover, if we roughly regard that $\frac{\partial^2 H(p\|q,\theta_k)}{\partial \theta_k \partial {\theta_k}^T}|_{\theta_k = \theta^o} = [NF(\theta^o)]^{-1}$ as N becomes large enough, we are alternatively lead to eq.(28)(b).

3.5 BYY Learning Integrates Regularization and Model Selection

Recall Sec. 2.1 and Sec. 2.2, the conventional regularization approaches have only a limited help on those learning problems due to a small sample size. Also, these regularization approaches suffer one crucial weakness caused by an isotropic regularization and a key difficulty on controlling a regularization strength. The conventional model selection approaches aim at tackling the weaknesses, but it suffers a huge cost to enumerate a number of candidate models with different values of k. Associated with a BYY system under the best harmony principle, the roles of regularization and model selection can be integrated in a sense that the crucial weakness caused by an isotropic regularization can be avoided by the model selection ability of the best harmony principle, while types of companion regularization can still be imposed to improve the weakness caused by the model selection mechanism of the BYY harmony learning. Moreover, some of these companion regularization approaches can also be decoupled from a BYY system and become directly applicable to the conventional maximum likelihood learning on a parametric model $p(x|\theta)$.

Considering $H(p\|q,\theta_k) = \int p_h(\mathcal{X}) H_{\mathcal{X}}(p\|q,\theta_k) d\mathcal{X} - Z(\theta_k)$ in eq.(19), we can also expand $H_{\mathcal{X}}(p\|q,\theta_k)$ with respect to $\mathcal{X}$ around $\bar{\mathcal{X}}_N$ up to the second order, resulting in

$$H(p\|q,\theta_k) = H_{\bar{\mathcal{X}}_N}(p\|q,\theta_k) + 0.5h^2 Tr[\frac{\partial^2 H_{\mathcal{X}}(p\|q,\theta_k)}{\partial \mathcal{X} \partial \mathcal{X}^T}]_{\mathcal{X}=\bar{\mathcal{X}}_N} - Z(\theta_k),$$
$$H_{\mathcal{X}}(p\|q,\theta_k) = \int p(\mathcal{Y}|\mathcal{X},\theta_k^{yx}) \ln [q(\mathcal{X}|\mathcal{Y},\theta_k^{xy}) q(\mathcal{Y}|\theta_k^y)] d\mathcal{Y}. \tag{29}$$

The term $0.5h^2 Tr[\cdot]$ is usually negative and thus increases as $h^2 \to 0$. What a value h will take depends on what type of the priori term $Z(\theta_k)$, for which there are three typical situations.

The simplest and also most usual case is $q(\theta_k) = 1$ or $Z(\theta_k) = 0$. In this case, $\max_h H(p\|q,\theta_k)$ will force $h = 0$, and thus we simply have $H(p\|q,\theta_k) = H_{\bar{\mathcal{X}}_N}(p\|q,\theta_k)$. When the function forms of $q(\bar{\mathcal{X}}_N|\mathcal{Y},\theta_k^{xy})$ and $q(\mathcal{Y}|\theta_k^y)$ are given while there is no priori constraint on the function form $p(\mathcal{Y}|\mathcal{X},\theta_k^{yx})$, we can consider the learning task in a sequential maximization, i.e., $\max_{\theta_k}\{\max_{p(\mathcal{Y}|\mathcal{X})} H(p\|q)\}$. It follows from $\max_{p(\mathcal{Y}|\mathcal{X})} H_{\mathcal{X}}(p\|q,\theta_k)$ that

$$p(\mathcal{Y}|\bar{\mathcal{X}}_N) = \delta(\mathcal{Y} - \bar{\mathcal{Y}}_N),\ \bar{\mathcal{Y}}_N = arg \max_{\mathcal{Y}} [q(\bar{\mathcal{X}}_N|\mathcal{Y},\theta_k^{xy}) q(\mathcal{Y}|\theta_k^y)],$$
$$H_{\bar{\mathcal{X}}_N}(p\|q,\theta_k) = H_{\bar{\mathcal{X}}_N,\bar{\mathcal{Y}}_N}(p\|q,\theta_k),$$
$$H_{\bar{\mathcal{X}}_N,\mathcal{Y}}(p\|q,\theta_k) = \ln [q(\bar{\mathcal{X}}_N|\mathcal{Y},\theta_k^{xy}) q(\mathcal{Y}|\theta_k^y)]. \tag{30}$$

That is, $\max_{\theta_k}\{\max_{p(\mathcal{Y}|\mathcal{X})} H(p\|q)\}$ becomes $\max_{\theta_k} H_{\bar{\mathcal{X}}_N,\bar{\mathcal{Y}}_N}(p\|q,\theta_k)$.

On one hand, the winner-take-all (WTA) maximization in eq.(30) indirectly implements a mechanism of selecting an appropriate value **k** that enables the automatic model selection discussed in Sec. 3.3. Readers are referred to Sec. 22.5 in [44], Sec. 23.3.2 in [45], Sec. II(B) in [42], and Sec. III(C) in [41]. However, there is also an other hand. If we subsequently implement $\max_{\theta_k} H_{\bar{\mathcal{X}}_N,\bar{\mathcal{Y}}_N}(p\|q,\theta_k)$ by ignoring the relation of $\bar{\mathcal{Y}}_N = arg \max_{\mathcal{Y}} [q(\bar{\mathcal{X}}_N|\mathcal{Y},\theta_k^{xy})$ $q(\mathcal{Y}|\theta_k^y)]$ to θ_k, we have to re-update $\max_{p(\mathcal{Y}|\mathcal{X})} H_{\mathcal{X}}(p\|q,\theta_k)$ by eq.(30). Though such an alternative maximization will gradually increase $H_{\mathcal{X}}(p\|q,\theta_k)$, it cannot avoid to get stuck in a local maximum or perhaps even a saddle point. Moreover, such an iterative maximization has to be made on the whole batch $\bar{\mathcal{X}}_N$ per step and thus is computationally very expensive. In an actual implementation [59, 53, 51, 52, 49, 47, 42, 43, 41], such an iteration is made per sample per step in a form $\bar{y}_t = arg \max_{y_t} [q(\bar{x}_t|y_t,\theta_k^{xy}) q(y_t|Y_{t-1},\theta_k^y)]$, where Y_{t-1} is either an empty set or a set that consists of a number of past samples of $y_{t-1}, \cdots, y_{t-p}$.

When $\bar{\mathcal{Y}}_N = arg \max_{\mathcal{Y}} [q(\bar{\mathcal{X}}_N|\mathcal{Y},\theta_k^{xy}) q(\mathcal{Y}|\theta_k^y)]$ becomes an explicit expression with respect to $\bar{\mathcal{X}}_N$ and θ_k or $\bar{y}_t = arg \max_{y_t} [q(\bar{x}_t|y_t,\theta_k^{xy}) q(y_t|Y_{t-1},\theta_k^y)]$ becomes an explicit expression with respect to x_t, Y_{t-1} and θ_k, we can take these explicit expressions into $\max_{\theta_k} H_{\bar{\mathcal{X}}_N,\bar{\mathcal{Y}}_N}(p\|q,\theta_k)$ by considering a compound dependence on θ_k, which will be helpful to improve the problem of local maximum. However, except for certain particular cases, it is usually difficult to get such explicit expressions. Therefore, we have to ignore the dependence of $\bar{\mathcal{Y}}_N$ with respect to θ_k.

The local maximum problem can be remedied by a unique regularization type coped with a BYY system, *structural regularization* or shortly BI-regularization. Details are referred to Sec. 3 in [46].

The key idea of the BI-regularization is to let $p(\mathcal{Y}|\mathcal{X})$ in an appropriately designed structure, three examples are given as follows:

(a) Let the optimization procedure for $\bar{\mathcal{Y}}_N$ in eq.(30) to be approximated by a parametric function:

$$\begin{aligned}&\bar{\mathcal{Y}}_N = F(\bar{\mathcal{X}}_N, \theta_k^{yx}),\ p(\mathcal{Y}|\bar{\mathcal{X}}_N) = \delta(\mathcal{Y} - \bar{\mathcal{Y}}_N),\\ &H_{\bar{\mathcal{X}}_N}(p\|q, \theta_k) = H_{\bar{\mathcal{X}}_N,\mathcal{Y}}(p\|q, \theta_k)_{\mathcal{Y}=F(\bar{\mathcal{X}}_N,\theta_k^{yx})},\end{aligned} \tag{31}$$

where two examples of $F(\bar{\mathcal{X}}_N, \theta_k^{yx})$ are as follows,
(1) $\bar{y}_t = f(x_t, \theta_k^{yx})$ for i.i.d. samples,
(2) $\bar{y}_t = f(x_t, \Xi_t, \theta_k^{yx})$ for temporal samples,
Ξ_t consists of past samples from either or both of $\bar{\mathcal{X}}_N$ and $\bar{\mathcal{Y}}_N$.

(b) We can also consider jointly a number of functions $\mathcal{Y} = F_\ell(\bar{\mathcal{X}}_N, \theta_{\ell,k}^{yx})$, $\ell = 1, \cdots, n_{yx}$ as follows

$$\begin{aligned}&\bar{\mathcal{Y}}_N = F_{\ell^*}(\bar{\mathcal{X}}_N, \theta_k^{yx}),\ \ell^* = arg\max_{\ell}[q(\bar{\mathcal{X}}_N|\mathcal{Y}, \theta_k^{xy})q(\mathcal{Y}|\theta_k^{y})]_{\mathcal{Y}=F_\ell(\bar{\mathcal{X}}_N,\theta_{\ell,k}^{yx})},\\ &H_{\bar{\mathcal{X}}_N}(p\|q, \theta_k) = H_{\bar{\mathcal{X}}_N,\mathcal{Y}}(p\|q, \theta_k)_{\mathcal{Y}=F_{\ell^*}(\bar{\mathcal{X}}_N,\theta_{\ell^*,k}^{yx})}.\end{aligned} \tag{32}$$

(c) In the special cases that $\mathcal{Y}$ is represented in discrete values or both $q(\bar{\mathcal{X}}_N|\mathcal{Y}, \theta_k^{xy})$ and $q(\mathcal{Y}|\theta_k^{y})$ are Gaussian, it is also possible to let

$$p(\mathcal{Y}|\mathcal{X}) = \frac{q(\bar{\mathcal{X}}|\mathcal{Y}, \theta_k^{xy})q(\mathcal{Y}|\theta_k^{y})}{\int q(\mathcal{X}|\mathcal{Y}, \theta_k^{xy})q(\mathcal{Y}|\theta_k^{y})d\mathcal{Y}}. \tag{33}$$

In these two cases, the integral over $\mathcal{Y}$ either becomes a summation or is analytically solvable. Readers are referred to Sec. 3.4.2 in [40] some discussions on the summation cases.

Another type of regularity lost comes from that $H_{\bar{\mathcal{X}}_N,\bar{\mathcal{Y}}_N}(p\|q, \theta_k)$ is computed only based on the samples in $\mathcal{X}_N$ via $p_0(\mathcal{X}_N) = p_h(\mathcal{X}_N)|_{h=0}$ by eq.(15). Though the above discussed structural regularization is helpful to remedy this problem indirectly, another regularization type coped with a BYY system is the Z-regularization featured by the term $Z(\theta_k) \neq 0$ in eq.(29), with two typical examples as follows:

- When $q(\theta_k)$ is irrelevant to h but relates to a subset of θ_k, $\max_h H(p\|q, \theta_k)$ will still force $h = 0$, and thus force the second term $0.5h^2Tr[\cdot]$ disappeared, while $Z(\theta_k)$ will affect the learning on θ_k. A typical example is

$$q(\theta_k^{xy}, \theta_k^{y}) \propto [\sum_{t=1}^{N} \int q(\bar{x}_t|y_t, \theta_k^{xy})q(y_t|\theta_k^{y})dy_t]^{-1}, \tag{34}$$

$$or\ q(\theta_k) = q(\theta_k^{xy}, \theta_k^{y}) \propto [\sum_{t=1}^{N} q(\bar{x}_t|\bar{y}_t, \theta_k^{xy})q(\bar{y}_t|\theta_k^{y})]^{-1},\ \text{if we get } \bar{y}_t \text{ per } \bar{x}_t,$$

which normalizes a finite size samples of $q(\bar{x}_t|y_t, \theta_k^{xy})q(y_t|\theta_k^y)$. Thus, it is called *normalization learning*. Readers are referred to Sec. 2.2 in [52], Sec. 22.6.1 in [44], and [47].

- Another typical case is $q(\theta_k) = q(h) \propto [\sum_{t=1}^N \sum_{\tau=1}^N G(x_t|x_\tau, h^2 I)/N]^{-1}$ that merely relates to h but is irrelevant to any other parameters in θ_k. In this case, the term $Z(\theta_k)$ together with the term $0.5h^2 Tr[\cdot]$ will trade off to give an appropriate h, which then affects the learning on other parameters in θ_k via $0.5h^2 Tr[\cdot]$. This type of regularization is equivalent to smoothing the samples in $\mathcal{X}_N$ via adding Gaussian noises with a noise variance h^2. Thus, it is called *data smoothing*. Details are referred to Sec. II(B) in [51], Sec. 23.4.1 in [45], and Sec. III(E) in [41].
  Furthermore, *data smoothing* can also be imposed on $\bar{\mathcal{Y}}_N$ given by eq.(31) or eq.(32) by considering

  $$\begin{aligned} & p_{h_y}(\mathcal{Y}|\bar{\mathcal{X}}_N) = G(\mathcal{Y}|\bar{\mathcal{Y}}_N, h_y^2 I), \\ & H_{\bar{\mathcal{X}}_N}(p\|q, \theta_k) = H_{\bar{\mathcal{X}}_N, \bar{\mathcal{Y}}_N}(p\|q, \theta_k) + 0.5h_y^2 Tr[\frac{\partial^2 H_{\bar{\mathcal{X}}_N, \mathcal{Y}}(p\|q, \theta_k)}{\partial \mathcal{Y} \partial \mathcal{Y}^T}]_{\mathcal{Y}=\bar{\mathcal{Y}}_N}, \\ & q(\theta_k) = q(h, h_y) \propto [\sum_{t=1}^N \sum_{\tau=1}^N G(x_t|x_\tau, h^2 I) G(y_t|y_\tau, h_y^2 I)/N]^{-1}. \end{aligned} \quad (35)$$

  In this case, the term $Z(\theta_k)$ together with the term $0.5h^2 Tr[\cdot] + 0.5h_y^2 Tr[\cdot]$ will trade off to give both h and h_y, which will affect the learning on other parameters in θ_k via $0.5h^2 Tr[\cdot] + 0.5h_y^2 Tr[\cdot]$.

Both the above Z-regularization approaches can be decoupled from a BYY system and become directly applicable to the conventional maximum likelihood learning on a parametric model $p(x|\theta)$, featured by their implementable ways for controlling regularization strength. For *data smoothing*, the regularization strength h^2 (equivalently the noise variance) is estimated in an easy implementing way. For *normalization learning*, the regularization strength is controlled by the term Z, with a conscience de-learning behavior. Readers are also referred to [71] for a rationale for the priors by eq. (34) and eq. (35).

In addition to all the above discussed, regularization can also be implemented by appropriately combining the best match principle and the best harmony principle. Readers are referred to Sec. 23.4.2 in [45] for a summary of several methods under the name KL-λ-HL spectrum, and also to [43] for the relations and differences of the best harmony principle from not only the best match principle but also several existing typical learning theories. In addition, the $\ln(r)$ function in eq.(19) can also be extended to a convex function $f(r)$, and a detailed discussion can be found in Sec. 22.6.3 of [44].

3.6 Best Harmony, Best Match, Best Fitting: BYY Learning and Related Approaches

The differences and relations between the BYY learning and several typical approaches have been discussed in [50, 43]. Here we further elaborate this issue via more clear illustrations in Fig. 4 and Fig. 5.

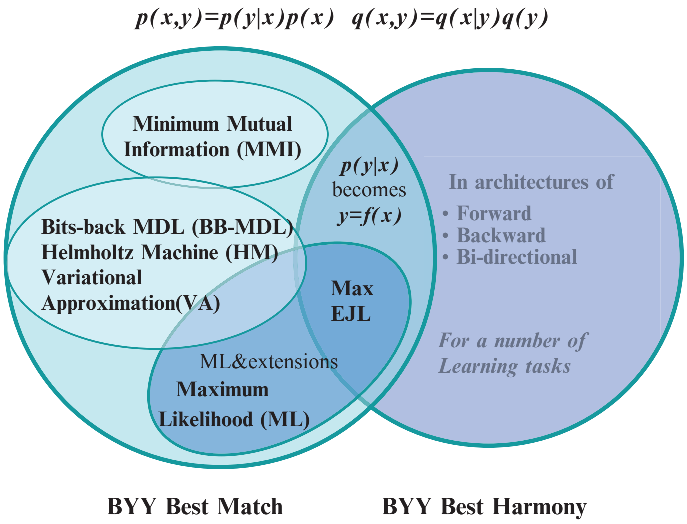


Fig. 4. BYY learning and related approaches (I)

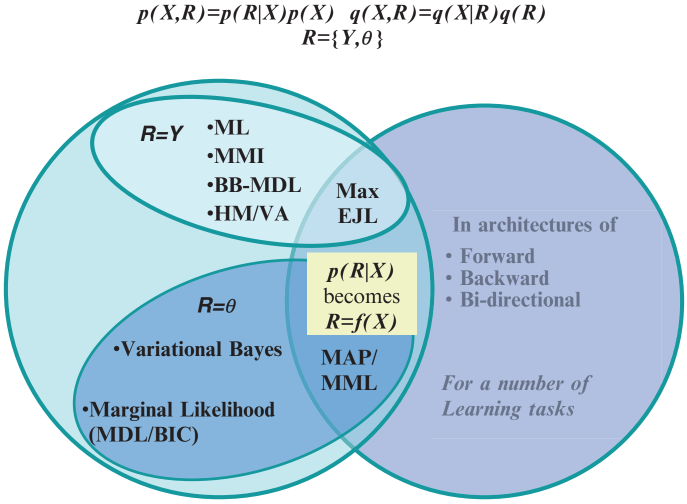


Fig. 5. BYY learning and related approaches (II)

As introduced in Sec. 3.1 and Sec. 3.2, learning in a BYY system can be implemented via either the best match principle by eq.(16) or the best harmony principle by eq.(18). The key difference of the best harmony principle by eq.(18) from the best match principle by eq.(16) is its model selection ability such that it can guide both parameter learning and model selection under a same principle, while the best match principle by eq.(16) can only guide parameter learning while model selection needs another different criterion.

This difference can be further understood in depth from a *Projection Geometry* Perspective (See Sec. III in [43]). The two principles correspond to two different types of *Geometry*, which become equivalent to each other only in a special case that $H(p\|p)$ is fixed at a constant H_0 that is irrelevant to both k and θ_k (see eqn(45) in [43]). For a BYY system, this case means that $p(\mathcal{Y}|\mathcal{X})$ in a format $p(\mathcal{Y}|\mathcal{X}) = \delta(\mathcal{Y} - f(\mathcal{X}))$ or $\mathcal{Y} = f(\mathcal{X})^{\dagger}$, as shown by the overlap part of two big circles in Fig. 4.

This common part has not been studied in the literature before it is studied under the best harmony principle in the BYY harmony learning on $H(p\|q, \theta_k)$ by eq.(18) with $Z = 1$. We say that this case has a backward architecture since only the backward direction $\mathcal{Y} \rightarrow \mathcal{X}$ has a specific structure while $p(\mathcal{Y}|\mathcal{X})$ is free from any structural constraint. In these cases, we get eq.(30) from eq.(19). Ignoring $-\ln Z$, it equivalently considers

$$\max_{\{\theta_k, k,\ \mathcal{Y}\}} \ln [q(\mathcal{X}_N|\mathcal{Y}, \theta_k^{xy})q(\mathcal{Y}|\theta_k^{y})]. \tag{36}$$

That is, we implement a best fitting of samples in $\mathcal{X}_N$ by a joint distribution via maximizing the extreme of the joint likelihood of $\mathcal{X}, \mathcal{Y}$ with respect to the unknown inner representation $\mathcal{Y}$. So, we call eq.(36) the maximum extremal joint likelihood or shortly Max-EJL, which will be further discussed in Sec. 5.2, especially after eq.(62).

In the BYY Best Match domain shown in Fig. 4, the other part of learning on a BYY system includes the widely studied Maximum Likelihood (ML), Minimum Mutual Information (MMI), the bits-back based MDL [13], Helmholtz Machine and Variational Approximation [10, 12]. The detailed discussions are referred to Sec. II in [43]. There are also other cases that have not been studied previously, e.g., *data smoothing* and *normalization learning*, which have already been introduced in Sec. 3.5.

Furthermore, we consider the most general case with $\mathcal{Y}$ replaced by $\mathcal{R} = \{\mathcal{Y}, \theta_k\}$ such that the best harmony principle is turned from eq.(18) into eq.(25), by which we got eq.(26), eq.(27), and eq.(29), as well as those studied in Sec. II(B) of [43].

Similarly, the best match principle is turned from eq.(16) into

$$D(p\|q) = \int p(\mathcal{R}|\mathcal{X})p(\mathcal{X})f(\frac{p(\mathcal{R}|\mathcal{X})p(\mathcal{X})}{q(\mathcal{X}|\mathcal{R})q(\mathcal{R})})d\mathcal{X}d\mathcal{R}, \tag{37}$$

which shares with eq.(25) a common part that $p(\mathcal{R}|\mathcal{X})$ in a format $p(\mathcal{R}|\mathcal{X}) = \delta(\mathcal{R} - f(\mathcal{X}))$ or $\mathcal{R} = f(\mathcal{X})^{\dagger}$, as shown by the overlap part of two big circles in Fig. 5. This common part includes not only *Max EJL* at $R = \mathcal{Y}$ but also what

[†] Strictly speaking, $H(p\|p)$ is still relevant to k even when $p(\mathcal{Y}|\mathcal{X})$ or $P(R|x)$ becomes a δ density, since it can be regarded as a limit of a type $-(c + \ln h)k$ as h → 0. In such a sense we should regard the above discussed common part in Fig. 4 and Fig. 5 is actually not common but only belongs to BYY Best Harmony domain, e.g., Max-EJL is a special case of BYY harmony learning only but not of BYY matching learning.

is usually called *MAximum Posterior (MAP)* at $R = \theta_k$, as well as closely relates to Minimum Message Length (MML) [36, 38].

In the *BYY best match* domain shown in Fig. 5, the other parts includes not only those listed in Fig. 4 at its special case $R = \mathcal{Y}$, but also the existing marginal likelihood related Bayesian approaches at $R = \theta_k$, such as MDL, BIC, and Variational Bayes.

4 Typical Examples

4.1 Gaussian Mixture

The first example that has been studied in [59] under the BYY harmony learning principle is the following Gaussian mixture, that is,

$$p(x|\theta_k) = \sum_{j=1}^{k} \alpha_j G(x|\mu_j, \Sigma_j). \tag{38}$$

In the literature, its parameters θ is learned via the maximum likelihood by the EM algorithm [25, 58], which has been widely studied in the literature of machine learning and neural networks. Readers are referred to Sec. 22.9.1(a) and Sec. 22.9.2(1) in [44] for a summary on a number of further results on the EM algorithm.

To determine k, we need one of typical model selection criteria such as AIC, BIC/MDL in help of a two-phase implementation. In [59], studies on the BYY learning for eq.(38) have been made in the common part shown in Fig. 4. On one hand, a criterion

$$J(k) = 0.5 \sum_{j=1}^{k} \alpha_j \ln |\Sigma_j| - \sum_{j=1}^{k} \alpha_j \ln \alpha_j, \tag{39}$$

and its special cases have been obtained [56]. On the other hand, an adaptive algorithm for implementing eq.(23) is firstly proposed in [59] under the name of the hard-cut EM algorithm for learning with automatic selection on k.

In eq.(38), x_t is a d-dimensional vector and y takes only discrete values $j = 1, \cdots, k$. When $p(\mathcal{X})$ is given by eq.(15), from eq.(18) we have

$$\begin{aligned} H(p\|q) &= \sum_{t=1}^{N} \sum_{j=1}^{k} \int p(j|x) G(x|x_t, h^2 I) \rho_j(x|\theta_j) dx, \\ \rho_j(x|\theta_j) &= \ln [\alpha_j G(x|\mu_j, \Sigma_j)]. \end{aligned} \tag{40}$$

In its simplest case that $h = 0$ and $Z = 1$, from eq.(21) we can get $J(k)$ in eq.(39) after discarding a constant $0.5d \ln (2\pi)$ and ignoring a term

$$\frac{0.5}{N} \sum_{j=1}^{k} Tr[\sum_{t=1}^{N} p(j|x_t)(x_t - \mu_j)(x_t - \mu_j)^T \Sigma_j^{-1}] = 0.5 \frac{d(N-k)}{N}. \tag{41}$$

When N is small, we can no longer regard $0.5d(N-k)/N$ as constant but need to consider $-0.5dk/N$. For eq.(27) and eq.(28), we have $d_k = dk+k-1+\sum_{j=1}^k d_{\Sigma_j}$, where d_A denotes the number of free parameters in the matrix A. It further follows that

$$\begin{aligned} J_C(k) &= 0.5\sum_{j=1}^k \alpha_j \ln|\Sigma_j| - \sum_{j=1}^k \alpha_j \ln \alpha_j - 0.5\frac{kd}{N}, \\ J_R(k) &= J_C(k) + c_k\frac{d_k}{N},\ d_k = dk+k-1+\sum_{j=1}^k d_{\Sigma_j}, \\ d_{\Sigma_j} &= \begin{cases} 1, & \text{for } \Sigma_j = \sigma_j^2 I, \\ d, & \text{for } \Sigma_j \text{ is diagonal,} \\ 0.5d(d+1), & \text{for } \Sigma_j \text{ in general,} \end{cases} \end{aligned} \tag{42}$$

where it follows from eq.(28) that

$$c_k = \begin{cases} 0.5, & \text{(a) corresponding to the case (a) in eq.(28),} \\ 1, & \text{(b) corresponding to the case (b) in eq.(28).} \end{cases} \tag{43}$$

Moreover, it follows from eq.(23) and eq.(40) that the updating formulae on $\alpha_j, \mu_j, \Sigma_j$ are same as their counterparts in the EM algorithm, while it follows from eq.(30) that

$$p(j|x_t) = \bar{\delta}_{jj_t^*},\ j_t^* = arg\max_j \rho_j(x_t|\theta_j),\ where\ \bar{\delta}_{ji} = \begin{cases} 1, & \text{if } j = i, \\ 0, & \text{otherwise,} \end{cases} \tag{44}$$

which is a hard-cut version of the posteriori probability of j upon x_t

$$p(j|x_t, \theta_j) = \frac{e^{-\rho_j(x_t|\theta_j)}}{\sum_{j=1}^k e^{-\rho_j(x_t|\theta_j)}}, \tag{45}$$

in the conventional EM algorithm. Thus, an algorithm that uses $p(j|x_t)$ in eq.(44) to replace the above one is named the hard-cut EM algorithm [59].

Generally, we consider to regularize learning on a small size N by

$$Z = \begin{cases} \frac{1}{N}\sum_{t=1}^N \sum_{\tau=1}^N G(x_t|x_\tau, h^2 I), & \text{(a) data smoothing with } h \neq 0, \\ \frac{1}{\sum_{t=1}^N \sum_{j=1}^k e^{\rho_j(x_t|\theta_j)}}, & \text{(b) normalization with } h = 0. \end{cases} \tag{46}$$

Interestingly, as shown in [52, 49], it follows from the above case (b) that we can get an algorithm from eq.(23) to demonstrate a mechanism similar to RPCL learning previously introduced in Sec. 2.3.

The type of bi-directional regularization by eq.(33) can also be imposed. E.g., as suggested by Eqn.(40) in [52], it follows from eq.(7) that we also get an algorithm that demonstrates another RPCL-like mechanism [19]. Actually, different types of bi-directional regularization demonstrate different versions of RPCL-like mechanism [46]. Readers are referred to Sec. 23.7 in [45] for a historic remark on RPCL-like mechanisms versus the BYY harmony learning, and to Eqn.(28) in [47] for a unified procedure to implement RPCL and adaptive EM as well as the hard-cut EM algorithm.

4.2 Local Subspaces and Local Factor Analysis

We further consider Σ_j in the following decomposition

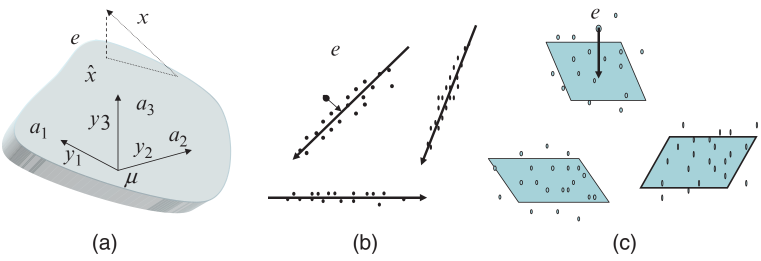
$$\Sigma_j = \sigma_j^2 I + \sum_{i=1}^{m_j} \lambda_j^{(i)\ 2} a_j^{(i)} a_j^{(i)\ T}, \ a_j^{(i)\ T} a_j^{(i)} = 1, \ a_j^{(i)\ T} a_j^{(\iota)} = 0, \ i \neq \iota, \quad (47)$$

where $\lambda_j^{(1)\ 2} \geq \lambda_j^{(2)\ 2} \cdots \geq \lambda_j^{(m_j)\ 2}$ with each $\lambda_j^{(i)\ 2}$ being the variance of the projection $a_j^{(i)\ T} x$ on the direction of the i-th principal vector $a_j^{(i)}$. This expression actually represents a subspace located at μ_j, shown in Fig. 6(a). Here, our task becomes finding several subspaces at different locations, which is thus called local subspace analysis. Readers are referred to Secs. 3.2 & 3.3 of [49] for not only other variants of elliptic RPCL but also RPCL based local subspaces.

When only the first principal component is considered, we can use this local PCA for collectively detecting multiple lines in an image, as shown in Fig. 6(b). We can also detect multiple planes in an image as shown in Fig. 6(c), as well as multiple curves and surfaces. Some applications are referred to [17, 18].

As illustrated in Fig. 6(a), the subspace obtained via the decomposition by eq.(47) is equivalent to orthogonally projecting each sample x onto a subspace that is located at μ and spanned by vectors a_1, a_2, a_3, such that the average square error $\|e\|^2$ between x and its projection $\hat{x}$ is minimized. It follows from $e = x - \hat{x}, \hat{x} = Ay + \mu$ that this subspace analysis is equivalent to the special case $\Sigma_j = \sigma_j^2 I$ of the following Factor Analysis (FA) [4]:

$$x = A_j y + \mu_j + e_j, \ e_j \sim G(e_j|0, \Sigma_j), \ y \sim G(y|0, I), \ E(e_j y^T) = 0, \quad (48)$$



(a) (b) (c)

Fig. 6. Subspaces

where $u \sim p(u)$ means that u comes from the distribution $p(u)$. In a general case $\Sigma_j \neq \sigma_j^2 I$, the project $x \to \hat{x}$ is still linear but its direction is no longer orthogonal to the subspace. Also, the average square error $\|e\|^2$ is no longer minimized.

Since a rotation transform on $y \sim G(y|0,I)$ results in $y' \sim G(y|0,I)$ still, it has no difference to $p(x)$ by eq.(48) whether A_j is a general matrix or subject to the following constraint

$$A_j = U_j \Lambda_j,\ U_j U_j^T = I,\ \Lambda_j^2 = diag[\lambda_j^{(1)\ 2}, \cdots, \lambda_j^{(m_j)\ 2}] \tag{49}$$

When $p(\mathcal{X})$ is given by eq.(15), putting eq.(48) into eq.(18) we have

$$\begin{aligned} H(p\|q) &= \sum_{t=1}^{N}\sum_{j=1}^{k} \int p(y|x,j)p(j|x)G(x|x_t,h^2 I)\rho_j(x,y|\theta_j)dxdy - \ln Z, \\ \rho_j(x,y|\theta_j) &= \ln[\alpha_j G(x|U_j\Lambda_j y + \mu_j, \Sigma_j)G(y|0,I)]. \end{aligned} \tag{50}$$

Similar to eq.(44), it follows from eq.(30) that

$$\begin{aligned} &p(y|x_t,j) = \delta(y - \hat{y}_j(x_t)),\ \hat{y}_j(x_t) = arg\max_y \rho_j(x_t,y|\theta_j), \\ &\hat{y}_j(x_t) = W_j(x_t - \mu_j),\ W_j = \Lambda_j U_j^T (U_j \Lambda_j^2 U_j^T + \Sigma_j)^{-1}. \end{aligned} \tag{51}$$

Also, we can get $p(j|x_t)$ by inserting $\rho_j(x_t|\theta_j) = \rho_j(x_t, \hat{y}_j(x_t)|\theta_j)$ into eq.(44). Again, $p(y|x_t,j)$ in eq.(51) can be regarded as a hard-cut version of the following posteriori probability of y upon getting x_t and j

$$p(y|x_t,j) = G(y|\hat{y}_j(x_t), I - \Pi_j),\ \Pi_j = W_j(U_j\Lambda_j^2 U_j^T + \Sigma_j)W_j^T. \tag{52}$$

In a way similar to eq.(39) and eq.(41), it follows from eq.(21) in the case $h = 0$ and $Z = 1$ that

$$\begin{aligned} &J(k, \{m_j\}_{j=1}^k) + c_{N,k} = 0.5\sum_{j=1}^{k} \alpha_j\{\ln|\Sigma_j| + J_j^y + m_j \ln(2\pi)\} - \sum_{j=1}^{k}\alpha_j \ln \alpha_j, \\ &J_j^y = \begin{cases} Tr[I - \Pi_j], & \text{(a) for } p(y|x_t,j) = G(y|\hat{y}_j(x_t), I - \Pi_j), \\ Tr[\Gamma_j], & \text{(b) for } p(y|x_t,j) = \delta(y - \hat{y}_j(x_t)), \end{cases} \\ &\text{where}\quad c_{N,k} = \frac{0.5}{N}(kd + \sum_{j=1}^{k} m_j), \\ &\Gamma_j = \frac{1}{\alpha_j N - 1}\sum_{t=1}^{N} p(j|x_t)\hat{y}_j(x_t)\hat{y}_j^T(x_t) = W_j S_j W_j^T, \\ &S_j = \frac{1}{\alpha_j N - 1}\sum_{t=1}^{N} p(j|x_t)(x_t - \mu_j)(x_t - \mu_j)^T. \end{aligned} \tag{53}$$

Specifically, the case (a) of J_j^y comes from $\int yy^T p(y|x_t,j)dy$, and the case (b) of J_j^y comes from $\sum_{j=1}^k \sum_{t=1}^N p(j|x_t)\hat{y}_j(x_t)\hat{y}_j^T(x_t)$, while $c_{N,k}$ comes in a way similar to eq.(41).

Moreover, similar to eq.(42) we have

$$J_R(k, \{m_j\}_{j=1}^k) = J(k, \{m_j\}_{j=1}^k) + c_k \frac{d_k}{N},$$
$$d_k = dk + k - 1 + \sum_{j=1}^k (m_j + d_{U_j} + d_{\Sigma_j}). \tag{54}$$

where c_k is same as in eq.(43), and $d_{U_j} = dm_j - 0.5m_j(m_j + 1)$.

Next, it follows from eq.(48) that the distribution $p(x)$ still remains unchanged when

$$A_j = U_j, \ y \sim G(y|0, \Lambda_j^2) \tag{55}$$

where the components of y remain uncorrelated. Correspondingly, eq.(50), eq.(51), eq.(52), and eq.(53) are modified with the following replacements

$$\rho_j(x, y|\theta_j) = \ln[\alpha_j G(x|U_j y + \mu_j, \Sigma_j) G(y|0, \Lambda_j^2)], \tag{56}$$
$$W_j = \Lambda_j^2 U_j^T (U_j \Lambda_j^2 U_j^T + \Sigma_j)^{-1},$$
$$p(y|x_t, j) = G(y|\hat{y}_j(x_t), \Lambda_j^2 - \Pi_j),$$
$$J_j^y = \begin{cases} \ln|\Lambda_j^2 - \Pi_j| + m_j, & \text{(a) for } p(y|x_t, j) = G(y|\hat{y}_j(x_t), \Lambda_j^2 - \Pi_j), \\ \ln|\Gamma_j| + m_j, & \text{(b) for } p(y|x_t, j) = \delta(y - \hat{y}_j(x_t)). \end{cases}$$

Still, we can get $p(j|x_t)$ by inserting the above $\rho_j(x_t|\theta_j) = \rho_j(x_t, \hat{y}_j(x_t)|\theta_j)$ into eq.(44), and also get $J_R(k, \{m_j\}_{j=1}^k)$ by eq.(54).

There are two specifical cases that deserve a particular mention. One is the special case $\Sigma_j = \sigma_j^2 I$ of eq.(48), which becomes local subspace analysis. The other is the special case $k = 1$, which becomes factor analysis. In the latter case, $J(k, \{m_j\}_{j=1}^k)$ and $J_R(k, \{m_j\}_{j=1}^k)$ are degenerated into $J(m_1)$ and $J_R(m_1)$ for determining the number of factors. Moreover, if we jointly have $k = 1$ and $\Sigma_1 = \sigma_1^2 I$, the problem becomes equivalent to *Principal Component Analysis (PCA)*, $J(m_1)$ and $J_R(m_1)$ can be used for determining the subspace dimension.

4.3 A Unified Learning Algorithm

On one hand, learning on the local factor analysis model by eq.(48) can be implemented in a two-phase implementation. That is, the first phase considers a set of possible candidate models by enumerating $k, \{m_j\}_{j=1}^k$ and then learns parameters in every candidate model in help of the EM algorithm under the maximum likelihood principle. The second phase selects a best candidate $k^*, \{m_j^*\}_{j=1}^{k^*}$ by either $J(k, \{m_j\}_{j=1}^k)$ or $J_R(k, \{m_j\}_{j=1}^k)$ given in the previous subsection. However, not only the computing cost will be impractically huge, especially for selecting $\{m_j\}_{j=1}^k$, but also this criterion type of multiple discrete variables becomes unable to provide a correct minimum point due to a finite sample size.

On the other hand, we can implement learning by eq.(23) in the cases of eq.(49) or eq.(55), during which $k^*, \{m_j^*\}_{j=1}^{k^*}$ is able to be automatically determined. For eq.(49), learning is made via eq.(50) during which m_j is determined via minimizing $-\ln G(y|0,I) = c + 0.5\|y\|^2$ that pushes $y^{(j)\ 2}$ of an extra dimension to 0. For eq.(55), learning is made via eq.(50) modified with eq.(56), during which m_j is determined via minimizing $-\ln G(y|0,\Lambda_j^2)$ that directly pushes each extra $\lambda_j^{(r)\ 2}$ to 0. From eq.(23), eq.(50), and eq.(56), we can obtain the detailed implementing algorithms. One example is the one given by Eqn.(72) plus Table 2 in [42] with $B_j = 0, \forall j$.

A general adaptive learning procedure is introduced in the sequel, which includes not only the one for implementing BYY harmony learning by eq.(23) in the form of eq.(50) and eq.(56) with $Z = 1$ and $h = 0$, but also an adaptive EM algorithm for the maximum likelihood learning, as well as a RPCL learning algorithm via the following rival penalized competition:

$$p_{j,t} = \begin{cases} 1, & \text{if } j = c, \quad c = arg\max_j \rho_j(x_t|\theta_j), \\ -\gamma, & \text{if } j = r, \quad r = arg\max_{j \neq c} \rho_j(x_t|\theta_j), \\ 0, & \text{otherwise}, \end{cases} \tag{57}$$

where $\rho_j(x_t|\theta_j)$ is either the one in eq.(40) or $\rho_j(x_t|\theta_j) = \rho_j(x_t, \hat{y}_j(x_t)|\theta_j)$ by eq.(50) or eq.(56).

The general procedure consists of iterating the following two steps:

Yang step: take a sample x_t, get $y_j(x_t) = W_j(x_t - \mu_j)$ with W_j by eq.(51) for eq.(49) or by eq.(56) for eq.(55). Then, with $\rho_j(x_t|\theta_j) = \rho_j(x_t, \hat{y}_j(x_t)|\theta_j)$ by eq.(50) or eq.(56), further get $p_{j,t}$ as follows

$$p_{j,t} = \begin{cases} p(j|x_t) \text{ by eq.(45)}, & \text{(a) ML} - \text{Learning}, \\ p(j|x_t,\theta_j) \text{ by eq.(44)}, & \text{(b) BYY harmony}, \\ \text{by eq.(57)}, & \text{(c) RPCL} - \text{Learning}, \\ p(j|x_t,\theta_j) - \gamma q(j|x_t), & \text{(d) RPCL} - \text{BYY harmony}, \end{cases} \tag{58}$$

where $\gamma > 0$ is a small number, and $q(j|x_t) \geq 0, \sum_{j=1}^k q(j|x_t) = 1$ are either pre-specified or estimated by some means, e.g., via a normalizing term Z.

Ying step: adaptively update all the parameters, check and discard extra dimensions. The details consist of

$$(a)\ \hat{x}_{j,t} = U_j^{old} y_{j,t} + \mu_j^{old},\ e_{j,t} = x_t - \hat{x}_{j,t},\ \mu_j^{new} = \mu_j^{old} + \eta p_{j,t} e_{j,t},$$

$$g_{U_j} = y_t e_{j,t}^T \Sigma_j^{old\ -1},\ U_j^{new} = U_j^{old} + \eta(g_{U_j}^T - U_j^{old} g_{U_j} U_j^{old}),$$

$$(b)\ \lambda_j^{(i)\ new} = \lambda_j^{(i)\ old} + \eta p_{j,t} \frac{y_{j,t}^{(i)\ 2} - (\lambda_j^{(i)\ old})^2}{\lambda_j^{(i)\ old}},$$

if $\lambda_j^{(i)} \to 0$ is detected, remove the i-th coordinate in the j-th subspace, $m_j \leftarrow m_j - 1$;

$$(c)\ \alpha_j = \frac{\beta_j^{2\ new}}{\sum_{j=1}^k \beta_j^{2\ new}},\ \beta_j^{new} = \beta_j^{old} + \eta \frac{p_{j,t} - \alpha_j^{old} \sum_{j=1}^k p_{j,t}}{\beta_j^{old}}, \tag{59}$$

if $\alpha_j \to 0$ is detected, discard the j-th subspace, $k \leftarrow k - 1$.

$$(d)\ \Sigma_j = S_j S_j^T,\ S_j^{new} = S_j^{old} + \eta p_{j,t} G_{\Sigma_j}^{old} S_j^{old},$$

$$G_{\Sigma_j} = \Sigma_j^{-1} e_{j,t} e_{j,t}^T \Sigma_j^{-1} - \Sigma_j^{-1}.$$

$$For\ \Sigma_j = \sigma_j^2 I,\ simply\ \sigma_j^{new} = \sigma_j^{old} + \eta p_{j,t} \frac{\|e_{j,t}\|^2/d - \sigma_j^{old\ 2}}{\sigma_j^{old}}.$$

The updating on U_j^{new} guarantees the satisfaction of $U_j U_j^T = I$. Moreover, even when $p_{j,t} < 0$, the updating rules (d)&(c) guarantee the satisfaction of $\alpha_j \geq 0, \sum_{j=1}^k \alpha_j = 1$ and that Σ_j remains non-negative definite.

4.4 Other Examples

The above general procedure degenerates back to the unified learning procedure by Eqn.(28) in [47] for Gaussian mixture by eq.(38) at $U_j = 0, \Lambda_j = I$. The above procedure also directly applies to the following two special cases:

- *Local subspace analysis* at the special case $\Sigma_j = \sigma_j^2 I$, which actually provides a unified scheme as well as improvements on the previous studies under the name of local PCA, local subspaces, and multi-sets mixture learning (MML) [62, 60]. Also, it can be applied to detecting lines, planes, curves, and surfaces in pattern recognition tasks [17, 18, 16].
- *Factor analysis* at the special case $k = 1$ that also includes principal components analysis (PCA). Readers are referred to Secs.2.2-2.4 in [48] and Sec. IV in [41] for recent summaries.

Advances on the BYY harmony learning have also been made along the following directions:

- *Independence subspace analysis* Extensions have been made from the specific case $G(y|0, \Lambda_j^2)$ to a general case $\prod_{j=1}^m q(y^{(j)})$, including independence components dependence (ICA), binary factor analysis (BFA), nonGaussian factor analysis (NFA), and LMSER, as well as three layer net. Readers are referred to Secs.4 & 5 in [48] and Sec. IV in [41].
- *Independence state space analysis* Extensions have further been made from $G(y|0, \Lambda_j^2)$ and $\prod_{j=1}^m q(y^{(j)})$ to temporal state spaces by taking temporal relations in consideration, including temporal factor analysis (TFA), independent hidden Markov model (HMM), temporal LMSER, and variants. Readers are referred to [42] and Sec. 6 in [48].
- *Mixture of shape-structures* In the computer vision field, finding multiple shapes is an important task called object detection. In [69], a new approach was proposed under the name of randomized Hough transform (RHT)

[65]. In [62, 60], a multi-set modelling method has been proposed, under situations of strong noise, partially observable objects, and a large amount of objects. In [40], a unified problem solving paradigm has been developed.

- *Combination of multiple inference* In [68], an early systematic study has been made on multiple classifier combination. In [63], a number of results have been obtained on statistical consistency and convergence rates for RBF nets. An alternative model of mixture of experts has been proposed and easily implemented by the EM algorithm [61], which is further applied to replace the existing suboptimal two stage algorithm for RBF nets [54]. Also, the number of basis functions are determined via either RPCL or BYY harmony learning. Readers are referred to Sec. 22.9.1(d) in [44].

5 A Trend and Challenges

5.1 A Trend for Model Selection

Summarizing the discussions on model selection in Sec. 2.2, Sec. 2.3, and Sec. 3, we roughly have two categories of studies. One is local cost based, usually for a learning task on a model that consists of several individual units or components. There is a local cost measure for each individual, e.g., $-\ln[\alpha_j G(x|\mu_j, \Sigma_j)]$ can be such a local cost for the j-th component in the Gaussian mixture by eq.(38). A sample x is excluded from one individual if its corresponding local cost is higher than a pre-specified threshold. If this x is excluded by all the current individual components, a new component is created to accommodate this x. As a result, a number of components are allocated to a set of samples. However, it is difficult to appropriately assign such a pre-specified threshold.

The other category is a global cost based, which is applicable to any model selection tasks. That is, after all its unknown parameters have been learned, a model with a scale k is globally evaluated by a cost $J(k)$ that is computed based on the learned parameters. Studies of this category can be further classified according to the configuration of $J(k)$.

When we have a large sample size N, as discussed in Sec. 1, a negative likelihood and a best fitting error, as well as a best matching error, will vary in a way illustrated by the top part of Fig. 7(a). That is, a correct scale k^* can be determined at the smallest k that either makes $J(k)$ reach its minimum or equivalently makes $\Delta J(k) = J(k+1) - J(k) = 0$. However, as the sample size N drops below a limit, this negative likelihood type $J(k)$ will degenerate into those shown by the top cases of Fig. 7(b) and Fig. 7(c). As a result, a correct scale k^* can not be obtained via searching a minimum or detecting a zero. One heuristic remedy is to check whether $\Delta J(k)$ is smaller than a pre-specified threshold. Alternatively, the conventional learning theories aim at modifying the top cases of Fig. 7(b) and Fig. 7(c) into the bottom cases

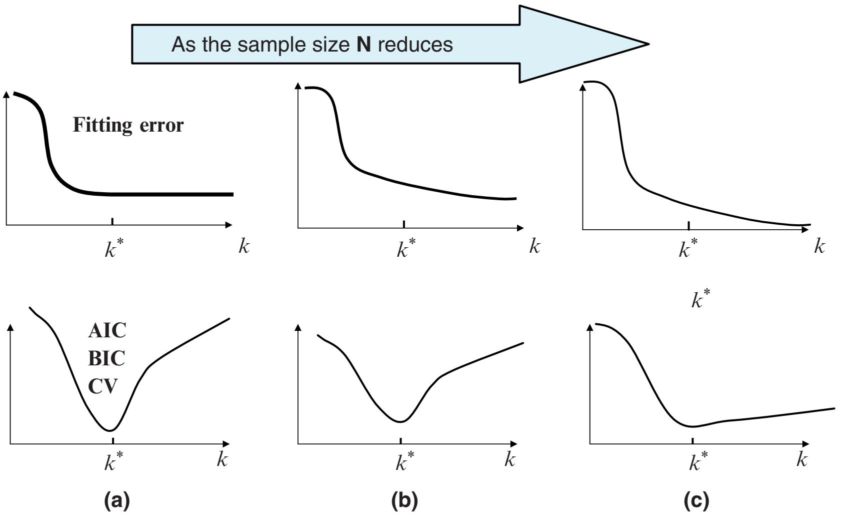


Fig. 7. Best matching error vs model selection criteria as the sample size N reduces

of Fig. 7(b) and Fig. 7(c) such that k^* can still be obtained via searching a minimum.

At each k, a cost $J(k)$ is computed once all the unknown parameters in the corresponding model have been estimated. Except for certain special task (e.g., determining the dimension k of orthogonal subspace), the estimated unknown parameters at one value of k usually can not be carried over to another value of k. More specifically, neither the estimated parameters at a lower value k' can be directly used as a part of the parameters at a higher value k'', nor the estimated parameters at a higher value k'' can be directly adopted for a use at a lower value k'. Therefore, all the unknown parameters have to be estimated completely at each different value k. That is, the model selection has to be implemented expensively.

On one hand, the BYY harmony learning provides us new criteria $J(k)$ by eq.(21), eq.(22), and eq.(27). Illustrated in the middle of Fig. 8 are those $J(k)$ curves by eq.(21) at different sample sizes of N, which are usually same or slightly worse than the popular model selection criterion BIC or equivalently MDL. The curve $J(k)$ by eq.(21) or eq.(22) are used as a bridge to illustrate the equivalent performance of automatic model selection by eq.(23) in a comparison with those existing conventional criteria, instead of being actually used in a two phase implementation for model selection. In a two phase implementation, it is suggested to use $J(k)$ by eq.(27), as illustrated in the bottom of Fig. 8, which are usually better than several typical criteria such as AIC, CAIC, BIC/MDL, CV, etc.

On the other hand, more important is that the BYY harmony learning provides a new trend that integrates model selection and parameter learning into one single process with a considerably reduced computing cost, that is,

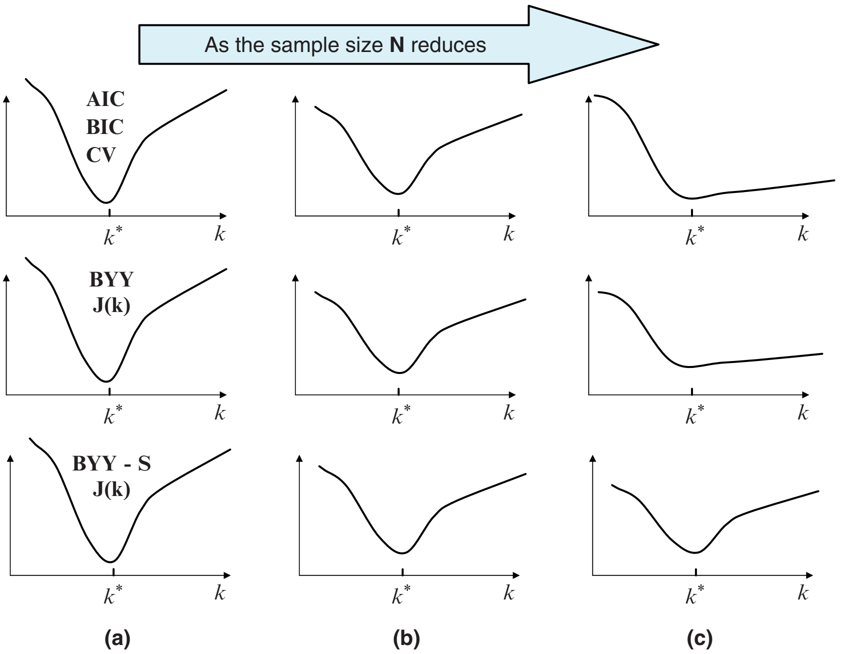


Fig. 8. BYY harmony learning criteria vs model selection criteria as the sample size N reduces

a trend of seeking automatic model selection during parameter learning. Further efforts are deserved along this trend. Generally speaking, in additional to the BYY harmony learning, an approach that follows this trend should be either explicitly or implicitly featured by a cost measure that varies with both the scale integer k and the parameters θ_k, in a format $f(\theta_k, k)$ or shortly $f(\theta_k)$ with the following nature:

$$f(\theta_k) \begin{cases} = f^*, & \text{when } k \geq k^*, \theta_{k^*} = \theta^*_{k^*} \text{ and } \phi_k = 0, \\ > f^*, & \text{otherwise,} \end{cases} \tag{60}$$

where $f^* = f(\theta^*_{k^*})$ is reached at the correct k^* and the correct value $\theta^*_{k^*}$. Moreover, $\theta_k = \{\theta_{k^*}, \theta^r_{k-k^*}\}$ with $\theta^r_{k-k^*}$ denoting the remaining part of θ_k after removing the subset θ_{k^*}, and ϕ_k is a critical subset $\phi_k \subseteq \theta^r_{k-k^*}$. For an example, we have $\phi_k = \{\alpha_j\}^k_{j=k^*+1}$ in eq.(38). When $\phi_k = 0$, a Gaussian mixture with k components actually becomes one with only k^* components.

Given a $k \geq k^*$ initially, minimizing $f(\theta_k)$ with respect to θ_k will force $\theta_{k^*} = \theta^*_{k^*}$ and $\phi_k = 0$, such that a model with a higher scale k actually becomes one with the correct k^* effectively. That is, model selection is made automatically during parameter learning. Moreover, it follows from eq.(60) that $J(k) = \min_{\theta_k} f(\theta_k)$ will illustrate as shown by Fig. 3(b) and that $J(k) = \min_{\theta_k,\ s.t.\ \phi'_k = c} f(\theta_k)$ will illustrate as shown by Fig. 3(a), where c is certain constant and $\phi'_k \supset \phi_k$ is a subset in θ_k, e.g., we have $c = 1/k$ and $\phi_k = \{\alpha_j\}_{j=1}^k$ in eq.(38).

5.2 Theoretical Issues in a Large Sample Size

Corresponding to the studies on asymptotic natures (i.e., the behaviors as the sample size $N \to \infty$) of the maximum likelihood approach or a best fitting type approach (e.g., the Kullback divergence based one by eq.(16)), it is also an interesting problem to study asymptotic natures of the BYY harmony learning. Such studies can be made in two stages.

First, we consider maximizing $H(p\|q, \theta_k)$ in eq.(18). Considering $f(r) = \ln r$ and that samples of x_t are i.i.d., we have $Z \to 1$ as $N \to \infty$ and thus eq.(18) becomes equivalent to the following form

$$H_o(p\|q, \theta_k) = \int p(y|x)p_o(x) \ln [q(x|y, \theta_k^{x|y})q(y, \theta_k^y)]\mu(dx)\mu(dy), \qquad (61)$$

where $p_o(x)$ is the original density that samples of x come from, k is one or a set of integers that represent the scale of y, and $\theta_k = \{\theta_k^{x|y}, \theta_k^y\}$ consists of the unknown parameters sets in the distribution functions $q(x|y, \theta_k^{x|y})$ and $q(y, \theta_k^y)$ respectively, with their structures pre-designed.

When $p(y|x)$ is free of any constraint, $\max_{p(y|x)} H_o(p\|q, \theta_k)$ results in

$$p(y|x) = \delta(y - y(x, \theta_k)),\ y(x, \theta_k) = arg \max_y [q(x|y, \theta_k^{x|y})q(y, \theta_k^y)], \qquad (62)$$
$$H_o(p\|q, \theta_k) = \int p_o(x) \ln Q(x, \theta_k)\mu(dx),\ Q(x, \theta_k) = q(x|y(x, \theta_k))q(y(x, \theta_k)),$$

which was previously discussed after eq.(36), as well as in Fig. 4 and Fig. 5, under the name Max-EJL.

In a comparison of the corresponding maximum likelihood counterpart, i.e.,

$$L_o(\theta_k) = \int p_o(x) \ln q(x, \theta_k)\mu(dx),$$
$$q(x, \theta_k) = \int q(x|y, \theta_k^{x|y})q(y, \theta_k^y)\mu(dy), \qquad (63)$$

we can observe that the above marginal integral $q(x, \theta_k)$ (i.e., a projected sum of $q(x|y, \theta_k^{x|y})q(y, \theta_k^y)$ to the domain of x) is replaced by $Q(x, \theta_k)$ in eq.(62) that is the peak point of $q(x|y, \theta_k^{x|y})q(y, \theta_k^y)$ in the domain of y per each fixed x. Illustratively, we can regard $q(x|y, \theta_k^{x|y})q(y, \theta_k^y)$ as a mountain in a $x - y$ coordinate system, $q(x, \theta_k)$ lumps the total sum of all the masses along the y-axis perpendicularly to the x-axis, while $Q(x, \theta_k)$ only places the mass on the highest ridge of the mountain perpendicularly to the x-axis. Noticing that the mountain is constrained to have a unit total mass, i.e., $\int q(x|y, \theta_k^{x|y})q(y, \theta_k^y)\mu(dx)\mu(dy) = 1$, maximizing $H_o(p\|q, \theta_k)$ will force the mountain shrink to concentrate swiftly to its highest ridge, while maximizing L_o can be achieved by those mountains with a unit mass that stretches along the y-axis in infinite many ways. This provides another perspective that explains why the BYY harmony learning has a model selection ability while the maximum likelihood learning has not.

Moreover, it follows from eq.(62) that

$$\begin{aligned} &H_o(p\|q,\theta_k) = \int p_o(x) \ln \tilde{Q}(x,\theta_k)\mu(dx) + C(\theta_k), \\ &\tilde{Q}(x,\theta_k) = Q(x,\theta_k)/C(\theta_k),\ C(\theta_k) = \int Q(x,\theta_k)\mu(dx), \end{aligned} \tag{64}$$

Maximizing $H_o(p\|q,\theta_k)$ not only forces the configuration of $q(x|y,\theta_k^{x|y})q(y,\theta_k^y)$ to shrink into its highest ridge for a largest $C(\theta_k)$, but also forces $\tilde{Q}(x,\theta_k)$ to match $p_o(x)$ as close as possible.

Comparing eq.(64) with eq.(63), we observe that the asymptotic nature of the BYY harmony learning can be investigated in two typical situations. First, when $C(\theta_k)$ is only relevant to k but irrelevant to θ_k. The asymptotic nature of the BYY harmony learning is similar to the asymptotic nature of the maximum likelihood learning. That is, the key point is to investigate the discrepancy between $p_o(x)$ and $Q(x,\theta_k)$ versus the discrepancy between $p_o(x)$ and $q(x,\theta_k)$. We consider the notations:

$$\begin{aligned} &\mathcal{P}_q(k) = \{q(x,\theta_k):\ for\ all\ \theta_k \in \Xi_k\}, \\ &\mathcal{P}_Q(k) = \{Q(x,\theta_k):\ for\ all\ \theta_k \in \Xi_k\}, \end{aligned} \tag{65}$$

which denote the distribution families that can be represented by $q(x,\theta_k)$ and $Q(x,\theta_k)$, respectively, where Ξ_k is the domain that θ_k takes values.

One typical asymptotic nature is the so called statistical consistency. For the maximum likelihood learning, having statistical consistency means that $q(x,\hat{\theta}_k) \to p_o(x)$ for a maximum likelihood estimator $\hat{\theta}_k$ as $N \to \infty$, or equivalently $p_o(x) \in \mathcal{P}_q(k)$, which is always possible when k becomes large enough. For the BYY harmony learning, when $C(\theta_k)$ is only relevant to k but irrelevant to θ_k, statistical consistency means $p_o(x) \in \mathcal{P}_Q(k)$, which is also possible when k become large enough.

It is interesting to further study the cases where a statistical consistency is not satisfied. Such cases are encountered either when k is not large enough or when $C(\theta_k)$ is relevant to θ_k. The following are several theoretical issues to be explored:

(a) When $C(\theta_k)$ is only relevant to k but irrelevant to θ_k, it deserves to investigate how the bias between $p_o(x)$ and $Q(x,\theta_k)$ and the bias between $p_o(x)$ and $q(x,\theta_k)$ vary as k in the cases with $N \to \infty$.

(b) With the unknown original density $p_o(x)$ replaced by the empirical density by eq.(8) in the above case (a), it deserves to further investigate how the bias between $p_o(x)$ and $q(x,\hat{\theta}_k)$ and the bias between $p_o(x)$ and $Q(x,\hat{\theta}_k)$ vary as k and N vary, where $\hat{\theta}_k$ is obtained by the maximum likelihood learning for $q(x,\hat{\theta}_k)$, and by the BYY harmony learning via eq.(19) or eq.(23) for $Q(x,\hat{\theta}_k)$.

(c) When $C(\theta_k)$ is relevant to θ_k, it follows from eq.(64) that the BYY harmony learning is somewhat similar to a Bayesian learning, with $C(\theta_k)$ taking a role similar to a priori, which usually introduces certain bias. It is interesting to investigate how the bias between $p_o(x)$ and $Q(x, \hat{\theta}_k)$ changes as k (or as both k and N, when $p_o(x)$ is replaced by eq.(8)).

(d) In the above cases, there are no regularization taking its role. As introduced in Sec. 3.5, several ways can be used for imposing certain regularization, and it deserves to study how the asymptotic nature of the BYY harmony learning is affected by regularization. Particularly, it deserves to investigate how the bias between $p_o(x)$ and $Q(x, \hat{\theta}_k)$ changes as k, N, and h, with $p_o(x)$ replaced by eq.(15). It also deserves to study how the bias between $p_o(x)$ and $Q(x, \hat{\theta}_k)$ changes as k and N, when $p(y|x)$ is free but in a structure as discussed in Sec. 3.5 (e.g., given by eq.(31) and eq.(32)).

(e) As discussed in [43], the maximum likelihood learning and the BYY harmony learning can be interpreted from two different views of geometry. It is interesting to further investigate the nature of manifold of $H(p\|q, \theta_k)$, as well as its relations to k, N, and h.

5.3 Challenges in a Small Sample Size

In the cases of a small sample size, we encounter more challenges on a number of theoretic and algorithmic aspects for not only the BYY harmony learning but also other model selection approaches. In the following, we discuss a number of typical challenges:

(a) One is to estimate $H(p\|q) = \int p(\theta_k|\mathcal{X})H(p\|q, \theta_k)d\theta_k$ by eq.(26) more accurately. It is approximated by $J(\mathbf{k}) = -H(p\|q, \theta_k^*) + 0.5d(\theta_k^*)$ in eq.(27) via considering a noninformative priori $q(\theta_k) = 1$ and a rough estimator $\theta_k^* = T(\bar{\mathcal{X}}_N)$ in help of the celebrated Cramer-Rao inequality. Interestingly, this $J(\mathbf{k})$ with $d(\theta_k^*)$ in the case (a) of eq.(28) can also be reached in help of using the idea of eq.(9) to estimate the bias

$$b(k, N) = E_{\mathcal{X}_N}\{E_{\mathcal{X}_N}[H(p\|q, \theta_k)]_{\theta_k=\theta_k^*}\} - E_{\mathcal{X}_N} H(p\|q, \theta_k^*). \tag{66}$$

Considering the case $H(p\|q, \theta_k) = H_{\bar{\mathcal{X}}_N, \bar{\mathcal{Y}}_N}(p\|q, \theta_k)$ given by eq.(30), i.e., $H(p\|q, \theta_k) = \ln[q(\bar{\mathcal{X}}_N|\bar{\mathcal{Y}}_N, \theta_k^{xy})q(\bar{\mathcal{Y}}_N|\theta_k^y)]$ that can be approximately regarded as the likelihood function jointly on $\bar{\mathcal{X}}_N, \bar{\mathcal{Y}}_N$ if we ignore the dependence of $\bar{\mathcal{Y}}_N$ on θ_k, we can directly get $b(k, N) = 0.5d_k/N$ from AIC [1, 2, 3]. That is, this road also leads to $J(\mathbf{k})$ by eq.(27) with $d(\theta_k^*)$ in the case (a) of eq.(28). Intuitively, the performance changing trend from the case (a) to the case (b) in eq.(28) will be somewhat similar to the changing trend from AIC to BIC [29, 15, 23]. A further improvement may be obtained via mathematical analysis on one $d(\theta_k^*)$ somewhere between the case (a) and the case (b) in eq.(28). It may also deserve to consider $q(\theta_k)$ in a priori distribution instead of simply setting $q(\theta_k) = 1$.

(b) In addition to making empirical comparisons with those typical model selection criteria discussed in Sec. 2.2, it remains to be challenges to make mathematical analysis on the chances and the magnitudes that k^* deviates from the correct one of the underlying distribution, with k^* given by eq.(21) and eq.(27) versus by those typical model selection criteria. Moreover, the studies on k^* by eq.(21) illustrate the performances of automatic model selection during parameter learning, while the studies on k^* by eq.(27) illustrate the performances in a two phase implementation. The performance gain by eq.(27) should also been evaluated together with its computation cost in a quantative way.

(c) More importantly, challenges lay on building up a mathematical link from the BYY harmony measure by eq.(27) or eq.(23) to the generalization error by eq.(4) either in a general sense or specifically for different learning tasks with different structures [41]. In other words, how to mathematically analyzes the performances of BYY harmony learning in the term of the generalization error with respect to the sample size N. A possible direction is to rewrite the BYY harmony measure in a format $\int p(\mathcal{X})R(\mathcal{X},k)\mu(d\mathcal{X})$, and then investigate it in an analogy of those studies on eq.(4). Similar challenges apply to those typical model selection criteria in Sec. 2.2 too.

(d) As introduced in Sec. 3.5, the BYY harmony learning integrates the roles of regularization and model selection. It is interesting to examine this feature in term of generalization error, i.e., how the generalization error is affected by regularization, especially how it varies as k, N, and h with $p_o(x)$ replaced by eq.(15), how it relates to $p(y|x)$ in a structure given by eq.(31) and eq.(32), as well as how accurate the scale k^* obtained by automatic model selection is, with respect to the sample size N and k^*.

(e) As discussed after eq.(30), alternatively making $\max_{\theta_k} H_{\bar{\mathcal{X}}_N,\bar{\mathcal{Y}}_N}(p\|q,\theta_k)$ and $\bar{y}_t = arg\max_{y_t}[q(\bar{x}_t|y_t,\theta_k^{xy})q(y_t|Y_{t-1},\theta_k^{y})]$ lead to the problem of local maximum or saddle point. It needs further investigation on how this problem affects the accuracy of the obtained scale k^* and the generalization error with its relation to k, N, and h, via either an implementation with automatic model selection or a two phase implementation.

(f) Also discussed after eq.(30), if we have an explicit expression for $\bar{y}_t = arg\max_{y_t}[q(\bar{x}_t|y_t,\theta_k^{xy})q(y_t|Y_{t-1},\theta_k^{y})]$, we can use it to improve the problem of local maximum. It deserves to study such an improvement in term of the accuracy of k^* and the generalization error. As introduced in Sec. 3.5, in the cases without such explicit expressions, one way to remedy is to approximate the desired explicit expression by a parametric structure, e.g., by eq.(31) and eq.(32). In this way, the relation of $\bar{\mathcal{Y}}_N$ or of $\bar{y}_t$ to θ_k can be approximately taken in consideration during updating θ_k. On the other hand, this pre-designed parametric structure may constrain $\max_{p(\mathcal{Y}|\mathcal{X})} H_{\mathcal{X}}(p\|q,\theta_k)$ to reach its optimal solution by eq.(30). Thus, it needs to examine the two-fold role of imposing such a pre-designed parametric structure.

(g) The difficulty of getting the above discussed explicit expression also brings an implementation difficulty. I.e., $\bar{\mathcal{Y}}_N = arg\max_{\mathcal{Y}}[q(\bar{\mathcal{X}}_N|\mathcal{Y}, \theta_k^{xy})q(\mathcal{Y}|\theta_k^{y})]$ or $\bar{y}_t = arg\max_{y_t}[q(\bar{x}_t|y_t, \theta_k^{xy})q(y_t|Y_{t-1}, \theta_k^{y})]$ has to be iteratively solved as an inner loop within a parameter learning process. It can be very expensive to wait for this iterative inner loop to converge. In practice, this convergence is approximately replaced by running the iteration only for a few steps, which can be far before its convergence. It would be also a challenge on appropriately developing such an approximation and on examining how it affects the performance.

(h) Related closely to the above case (f) and case (g), it would be helpful to investigate the manifold of $H_{\mathcal{X}}(p\|q, \theta_k)$, especially on the distribution of local maxima. Since the gradient based technique takes a major role in implementing the BYY harmony learning, it is likely to be stuck at a local maximum. Thus, it deserves to study how the global versus local maximum issue will affect the accuracy of k^* and the generalization error.

(i) As introduced in Sec. 2.3, Rival Penalized Competitive Learning (RPCL) can also perform automatic model selection [64, 66]. Its relation to the BYY harmony learning has been discussed in [52, 49, 47, 19]. Though the convergence behavior of RPCL has already been qualitatively interpreted in a top-down way via the BYY harmony learning (e.g., as discussed in Sec. 4.1), it is interesting to investigate in a bottom up way on converging behaviors of both RPCL learning and adaptive algorithm for BYY harmony learning. Some preliminary studies have been made in [20, 21]. However, challenges still remain on getting the conditions (especially the penalizing strength) for guaranteeing a RPCL learning to correctly converge with a correct scale k^*.

(j) As discussed in Sec. 5.1, the key point of automatic model selection comes from the nature by eq.(60). That is, a critical subset ϕ_k of parameters, e.g., the proportional parameter α_j in eq.(38) and one component variance $\lambda_j^{(r)\ 2}$ of Λ_j^2 in eq.(55), will be driven towards 0 during the maximization process. However, waiting for ϕ_k converging to zero will waste a large computing cost, which is usually unnecessary. It deserves to develop effective techniques to detect the evidences for $\phi_k \to 0$ (e.g., $\alpha_j \to 0$, $\lambda_j^{(r)\ 2} \to 0$). One possible direction is to develop some statistical test for this purpose.

(k) The BYY harmony learning has already been extended to model temporal relations among samples [51, 48]. Many of the above discussed challenges should also be investigated on these temporal extensions.

(l) In a two-stage implementation given at the end of Sec. 2.2, we have to enumerate every k within $[k_d, k_u]$ in a general case without considering the internal structure of $J(k, \theta_k^{fit})$. Such an enumeration can be made in either a forward way (i.e., increasing k from a small initial value) or a backward way (i.e., decreasing k from a large initial value). In a forward implementation, as k increases to $k + 1$, not only those newly appeared parameters but also the existing parameter set θ_k have to be re-learned.

In a backward implementation, as $k+1$ decreases to k, we are unable to directly take a subset θ_k from the set θ_{k+1}. However, $J(k, \theta_k^{fit})$ may have a specific structure for certain learning tasks. As k increases to $k+1$, only those newly appeared parameters need to be re-learned while the existing parameter set θ_k remain unchanged. In other words, the learning can be made in an incremental way. Furthermore, we may also consider $J(k, \theta_k^{fit})$ with a more complicated structure in help of certain enumerating or searching technique that was developed for feature selection tasks in the pattern recognition field [70].

6 Concluding Remarks

Advances, trends, and challenges on regularization and model selection in statistical learning have been discussed from a Bayesian Ying Yang learning perspective. After briefly introducing the Bayesian Ying-Yang system and the best harmony learning principle, its advantage of automatic model selection and of integrating regularization and model selection have been addressed, and its differences and relations to typical existing learning methods have been elaborated. Taking the tasks of Gaussian mixture, local subspaces, local factor analysis as examples, not only detailed model selection criteria are given, but also a general learning procedure is provided to unify adaptive algorithms for these learning tasks. Finally, a new trend for model selection has been elaborated; theoretical issues in a large sample size and challenges in a small sample size have been further presented.

Acknowledgement

The work described in this paper was fully supported by a grant from the Research Grant Council of the Hong Kong SAR (Project No:CUHK4173/06E).

References

[1] Akaike, H (1974), "A new look at the statistical model identification", *IEEE Tr. Automatic Control*, 19, 714-723.

[2] Akaike, H (1981), "Likelihood of a model and information criteria", *Journal of Econometrics*, 16, 3-14.

[3] Akaike, H (1987), "Factor analysis and AIC", *Psychometrika*, 52, 317-332.

[4] Anderson, TW, & Rubin, H (1956), "Statistical inference in factor analysis", *Proc. Berkeley Symp. Math. Statist. Prob. 3rd 5*, UC Berkeley, 111-150.

[5] Bishop, C.M., (1995), "Training with noise is equivalent to Tikhonov regularization", *Neural Computation 7*, 108-116.

[6] Bozdogan, H (1987) “Model Selection and Akaike’s Information Criterion: The general theory and its analytical extension”, *Psychometrika*, **52**, 345-370.

[7] Bozdogan, H & Ramirez, DE (1988), “FACAIC: Model selection algorithm for the orthogonal factor model using AIC and FACAIC”, *Psychometrika*, 53 (3), 407-415.

[8] Cavanaugh, JE (1997), “Unifying the derivations for the Akaike and corrected Akaike information criteria”, *Statistics & Probability Letters*, 33, 201-208.

[9] Cooper, G & Herskovitz, E (1992), “A Bayesian method for the induction of probabilistic networks from data”, *Machine Learning*, 9, 309-347.

[10] Dayan, P. & Hinton, GE (1995), “The Helmholtz machine”, *Neural Computation 7*, No.5, 889-904.

[11] Girosi, F, et al, (1995) “Regularization theory and neural architectures”, *Neural Computation*, 7, 219-269.

[12] Hinton, GE, Dayan, P, Frey, BJ, & Neal, RN (1995), “The wake-sleep algorithm for unsupervised learning neural networks”, *Science 268*, 1158-1160.

[13] Hinton, GE & Zemel, RS (1994), “Autoencoders, minimum description length and Helmholtz free energy”, *Advances in NIPS*, 6, 3-10.

[14] Hurvich, CM, & Tsai, CL (1989), “Regression and time series model in samll samples”, *Biometrika*, 76, 297-307.

[15] Kashyap, RL (1982), “Optimal choice of AR and MA parts in autoregressive and moving-average models”, *IEEE Trans. PAMI*, 4, 99-104.

[16] Z.Y. Liu, H. Qiao, & L. Xu, “Multisets Mixture learning based Ellipse Detection”, *Pattern Recognition 39*, pp 731-735, 2006.

[17] Z.Y. Liu, K.C. Chiu, & L. Xu, “Strip Line Detection and Thinning by RPCL-Based Local PCA”, *Pattern Recognition Letters* **24**, 2335-2344, 2003.

[18] Liu, ZY, Chiu, KC, & Xu, L (2003), “ Improved system for object detection and star/galaxy classification via local subspace analysis”, *Neural Networks 16*, 437-451.

[19] Ma, J, Wang, T, & Xu, L (2004), “A gradient BYY harmony learning rule on Gaussian mixture with automated model selection”, *Neurocomputing 56*, 481-487.

[20] Ma, J & Xu, L (2002), “Convergence Analysis of Rival Penalized Competitive Learning (RPCL) Algorithm”, *Proc. of Intl. Joint Conf. on Neural Networks (IJCNN ’02)*, Hawaii, USA, May 12-17, 2002, pp 1596-1602.

[21] Ma, J & Xu, L “The Correct Convergence of the Rival Penalized Competitive Learning (RPCL) Algorithm”, *Proc. of Intl. Conf. on Neural Information Processing (ICONIP’98)*, October 21-23, 1998, Kitakyushu, Japan, Vo.1, pp239-242.

[22] Mackey, D (1992) “A practical Bayesian framework for backpropagation”, *Neural Computation*, 4, 448-472.

[23] Neath, AA & Cavanaugh, JE (1997), "Regression and Time Series model selection using variants of the Schwarz information criterion", *Communications in Statistics A*, 26, 559-580.

[24] T.Poggio & F.Girosi, "Networks for approximation and learning", *Proc. of IEEE*, **78**, 1481-1497 (1990).

[25] Redner, RA & Walker, HF (1984), "Mixture densities, maximum likelihood, and the EM algorithm", *SIAM Review*, 26, 195-239.

[26] Rissanen, J (1986), "Stochastic complexity and modeling", *Annals of Statistics*, 14(3), 1080-1100.

[27] Rissanen, J (1989), *Stochastic Complexity in Statistical Inquiry*, World Scientific: Singapore.

[28] Rivals, I & Personnaz, L (1999) "On Cross Validation for Model Selection", *Neural Computation*, 11, 863-870.

[29] Schwarz, G (1978), "Estimating the dimension of a model", *Annals of Statistics*, 6, 461-464.

[30] Stone, M (1974), "Cross-validatory choice and assessment of statistical prediction", *J. Royal Statistical Society B*, **36**, 111-147.

[31] Stone, M (1977), "An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion", *J. Royal Statistical Society B*, 39 (1), 44-47.

[32] Stone, M (1978), "Cross-validation: A review", *Math. Operat. Statist.*, 9, 127-140.

[33] Stone, M (1979), "Comments on model selection criteria of Akaike and Schwartz. *J. Royal Statistical Society B*, **41** (2), 276-278.

[34] Sugiura, N (1978), "Further analysis of data by Akaike's information criterion and the finite corrections", *Communications in Statistics A*, **7**, 12-26.

[35] Tikhonov, AN & Arsenin, VY (1977), *Solutions of Ill-posed Problems*, Winston and Sons.

[36] Wallace, CS & Boulton, DM (1968), "An information measure for classification", *Computer Journal*, 11, 185-194.

[37] Wallace, CS & Freeman, PR (1987), "Estimation and inference by compact coding", *J. of the Royal Statistical Society*, 49(3), 240-265.

[38] Wallace, CS & Dowe, DR (1999), "Minimum message length and Kolmogorov complexity", *Computer Journal*, 42 (4), 270-280.

[39] Vapnik, VN (1995), *The Nature Of Statistical Learning Theory*, Springer.

[40] Xu, L., (2007), "A Unified Perspective and New Results on RHT Computing, Mixture Based Learning, and Multi-learner Based Problem Solving", *Pattern Recognition,* Vol. 40, pp. 2129–2153, 2007.

[41] Xu, L., (2005), "Fundamentals, Challenges, and Advances of Statistical Learning for Knowledge Discovery and Problem Solving: A BYY Harmony Perspective", Keynote talk, *Proc. of Intl. Conf. on Neural Networks and Brain*, Oct. 13-15, 2005, Beijing, China, Vol. 1, pp. 24-55.

[42] Xu, L. (2004), “Temporal BYY Encoding, Markovian State Spaces, and Space Dimension Determination”, *IEEE Tr. Neural Networks, V15, N5*, pp. 1276-1295.

[43] Xu, L (2004), “Advances on BYY harmony learning: information theoretic perspective, generalized projection geometry, and independent factor auto-determination”, *IEEE Tr. Neural Networks, V15, N4*, pp. 885-902.

[44] Xu, L. (2004), “Bayesian Ying Yang Learning (I): A Unified Perspective for Statistical Modeling”, *Intelligent Technologies for Information Analysis*, N. Zhong and J. Liu (eds), Springer, pp. 615-659.

[45] Xu, L. (2004), “Bayesian Ying Yang Learning (II): A New Mechanism for Model Selection and Regularization”, *Intelligent Technologies for Information Analysis*, N. Zhong and J. Liu (eds), Springer, pp. 661-706.

[46] Xu, L. (2004), “BI-directional BYY Learning for Mining Structures with Projected Polyhedra and Topological Map”, Invited talk, in *Proc. of FDM 2004: Foundations of Data Mining*, eds., T.Y.Lin, S.Smale, T. Poggio, and C.J. Liau, Brighton, UK, Nov. 01, 2004, pp. 5-18.

[47] Xu, L. (2003), “Data smoothing regularization, multi-sets-learning, and problem solving strategies”, *Neural Networks, V. 15, No. 5-6*, 817-825.

[48] Xu, L. (2003), “Independent Component Analysis and Extensions with Noise and Time: A Bayesian Ying-Yang Learning Perspective”, *Neural Information Processing Letters and Reviews*, Vol.1, No.1, 1-52.

[49] Xu, L (2002), “BYY Harmony Learning, Structural RPCL, and Topological Self-Organizing on Mixture Models ”, *Neural Networks, V15, N8-9*, 1125-1151.

[50] Xu, L, (2002), “Bayesian Ying Yang Harmony Learning”, *The Handbook of Brain Theory and Neural Networks*, Second edition, (MA Arbib, Ed.), Cambridge, MA: The MIT Press, pp. 1231-1237.

[51] Xu, L (2001), “BYY Harmony Learning, Independent State Space and Generalized APT Financial Analyses ”, *IEEE Tr. Neural Networks*, **12** (4), 822-849.

[52] Xu, L (2001), “Best Harmony, Unified RPCL and Automated Model Selection for Unsupervised and Supervised Learning on Gaussian Mixtures, Three-Layer Nets and ME-RBF-SVM Models”, *Intl J of Neural Systems* **11** (1), 43-69.

[53] Xu, L (2000), “Temporal BYY Learning for State Space Approach, Hidden Markov Model and Blind Source Separation”, *IEEE Tr. Signal Processing 48*, 2132-2144.

[54] Xu, L (1998), “RBF Nets, Mixture Experts, and Bayesian Ying-Yang Learning”, *Neurocomputing*, Vol. 19, No.1-3, 223-257.

[55] Xu, L, (1998), “Rival Penalized Competitive Learning, Finite Mixture, and Multisets Clustering”, *Proc. of IJCNN98*, Anchorage, Vol.II, pp. 2525-2530.

[56] Xu, L (1997), “Bayesian Ying-Yang Machine, Clustering and Number of Clusters”, *Pattern Recognition Letters 18*, No.11-13, 1167-1178.

[57] Xu, L, (1997), “New Advances on Bayesian Ying-Yang Learning System with Kullback and Non-Kullback Separation Functionals”, *Proc. IEEE-INNS Intl. Joint Conf. on Neural Networks (IJCNN97)*, Houston, Vol. III, pp. 1942-1947.

[58] Xu, L, & Jordan, MI (1996), “On convergence properties of the EM algorithm for Gaussian mixtures”, *Neural Computation, 8*, No.1, 1996, 129-151.

[59] Xu, L, (1995), “Bayesian-Kullback Coupled YING-YANG Machines: Unified Learnings and New Results on Vector Quantization”, *Proc. Intl. Conf. on Neural Information Processing*, Oct 30-Nov.3, 1995, Beijing, pp. 977-988.

[60] L. Xu, “A Unified Learning Framework: Multisets Modeling Learning”, Invited Talk, *Proc. of World Congress on Neural Networks (WCNN95)*, Washington, DC, July 17-21, 1995, Vol.I, pp. 35-42.

[61] Xu, L, Jordan, MI, & Hinton, GE (1995), “An Alternative Model for Mixtures of Experts”, *Advances in Neural Information Processing Systems 7*, eds, Cowan, JD, et al, MIT Press, 633-640, 1995.

[62] L. Xu, “Multisets Modeling Learning: An Unified Theory for Supervised and Unsupervised Learning”, Invited Talk, *Proc. of IEEE ICNN94*, Orlando, Florida, June 26-July 2, 1994, Vol.I, 315-320.

[63] Xu, L, Krzyzak, A, & Yuille, AL (1994), “On Radial Basis Function Nets and Kernel Regression: Statistical Consistency, Convergence Rates and Receptive Field Size”, *Neural Networks*, **7**, 609-628.

[64] Xu, L, Krzyzak, A & Oja, E (1993), “Rival Penalized Competitive Learning for Clustering Analysis, RBF net and Curve Detection”, *IEEE Tr. on Neural Networks 4*, 636-649.

[65] Xu, L & Oja, E. (1993), “Randomized Hough Transform (RHT): Basic Mechanisms, Algorithms and Complexities”, *Computer Vision, Graphics, and Image Processing : Image Understanding*, Vol.57, No.2, pp. 131-154.

[66] Xu, L, Krzyzak, A & Oja, E (1992), “Unsupervised and Supervised Classifications by Rival Penalized Competitive Learning”, *Proc. of 11th Intl Conf. on Pattern Recognition (ICPR92)*, Hauge, Netherlands, Vol.I, pp. 672-675.

[67] Xu, L, Klasa, A, & Yuille, A.L. (1992), “Recent Advances on Techniques Static Feedforward Networks with Supervised Learning”, *International Journal of Neural Systems*, Vol.3, No.3, pp. 253-290.

[68] Xu, L., Krzyzak, A., & Suen, C.Y. (1992), “Several Methods for Combining Multiple Classifiers and Their Applications in Handwritten Character Recognition”, *IEEE Tr. System, Man and Cybernetics*, Vol. 22, No.3, pp. 418-435.

[69] Xu, L, Oja, E., & Kultanen, P. (1990), “A New Curve Detection Method: Randomized Hough Transform (RHT)”, *Pattern Recognition Letters*, Vol.11, pp. 331-338.

[70] Xu, L, P.F. Yan, & T. Chang (1988), “Best First Strategy for Feature Selection”, *Proc. of 9th Intl Conf. on Pattern Recognition (ICPR98)*, Nov. 14-17, 1988, Rome, Italy, Vol.II, pp. 706-709.

[71] Xu, L, (2007), “Bayesian Ying Yang Learning”, *Scholarpedia,* p. 10469, http://scholarpedia.org/article/Bayesian_Ying_Yang_Learning.