Proceedings of International Conference on Neural Networks and Brain, Vol. 1, pp.24-55, Oct. 13-15, 2005, Beijing, China.

Fundamentals, Challenges, and Advances of Statistical Learning for Knowledge Discovery and Problem Solving: A BYY Harmony Perspective

(Keynote Speech)

Lei Xu, Department of Computer Science and Engineering, Chinese University of Hong Kong,
Shatin, NT, Hong Kong, P.R. China, E-mail: lxu@cse.cuhk.edu.hk

***Abstract*—In this paper[1], an intelligent system is featured by both its abilities of interpreting what are observed via discovering knowledge about the world it survives, and its problem solving skills of handling each issue encountered in the world. Correspondingly, the abilities and skills are obtained by two types of learning via evidences or data from the world. Due to noises in observation and a finite size of samples, learning is statistical in nature, which faces two key challenges. One is finding appropriate mathematical representations to suit various dependence structures underlying world. The other is getting a good theory to guide learning such that dependence structures are not only learned into mathematical representations but also with an appropriate complexity that matches the size of samples (i.e., learning reliable structures of underlying world). This paper consists of part parts. The first two parts summarize typical dependence structures for tackling the challenge one and typical learning theories for tackling for tackling the challenge two. The third part introduces Bayesian Ying Yang (BYY) system as a general framework that unifies typical dependence structures and BYY harmony learning for the challenge two, with several favorable features. To illustrate this BYY learning, in the fourth part we further introduce fundamentals of independence subspaces and advances obtained from BYY harmony learning on typical independence subspaces, including PCA, MCA, DCA, ICA, FA, TFA, NFA, BFA, LMSER, as well as their temporal extensions. Finally, a concluding remark is made and new results of BYY learning in other learning areas are also briefly listed.**

I. Statistical Learning for Knowledge Discovery and Problem Solving

A. *Intelligent abilities and two types of learning*

An intelligent system, which could be an individual or a collection of men, animals, robots, agents, and other intelligent bodies, survives in its world with needs of two types of intelligent abilities.

As illustrated by the right path in Fig.1, Type-I consists of abilities of knowing 'what' or discovering its world, i.e., mining among data or information from things and events it has encountered and discovering regularities or dependencies among data as its knowledge about the world. As illustrated in Fig.2, the knowledge is obtained either from a huge volume of existing authorized sources (e.g., textbooks) or from pieces of uncertain evidences (or called samples) that directly come from the world during the activities, such as observation, experiments, exploration, think, communication, and collaboration as shown in Fig.3. Actually those authorized sources were also obtained from samples in past. Therefore, in its nature, Type-I abilities are obtained via processes what we usually call *learning*, during which the intelligent system gradually senses its world from samples and modifies itself to adapt the world. For this reason, we may also call an intelligent system shortly by a *learner*.

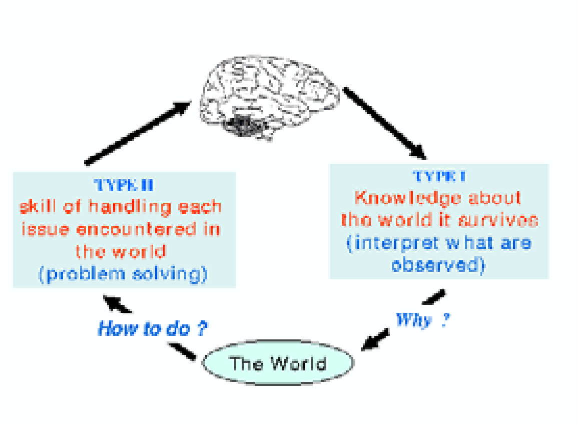

TYPE II
skill of handling each issue encountered in the world
(problem solving)
TYPE I
Knowledge about the world it survives
(interpret what are observed)
How to do ?
Why ?
The World

Fig. 1. Two Types of Intelligent Ability

As illustrated by the left path in Fig.1, Type-II consists of skills of problem solving, i.e., skills of appropriately reacting upon things and events that it is currently encountering. The reaction can be either just perceiving (e.g., identify, recognize, etc) the things and events or further making feedback actions to satisfy certain motivation (e.g., driving, operation, cooperation, competition, etc), as shown in Fig.4. These reactions are featured by rapid responses that demand a fast implementation mechanism trained (or called learned) from samples during these activities. As illustrated in Fig.2, the skills of problem solving may also be obtained in help

[1] The work described in this paper was fully supported by a grant from the Research Grant Council of the Hong Kong SAR (project No: CUHK4225/04E).

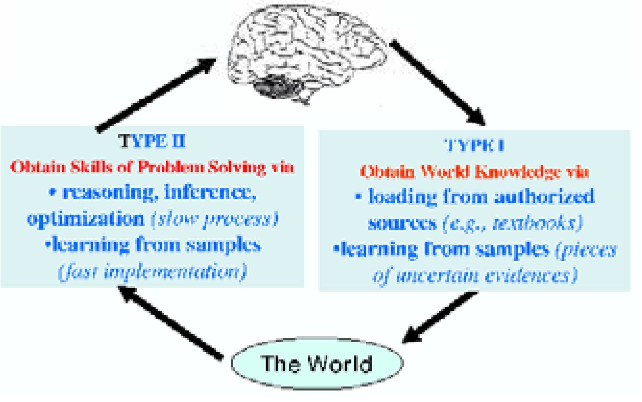

TYPE II
Obtain Skills of Problem Solving via
• reasoning, inference, optimization (slow process)
•learning from samples (fast implementation)
TYPE I
Obtain World Knowledge via
• loading from authorized sources (e.g., textbooks)
•learning from samples (pieces of uncertain evidences)
The World

Fig. 2. How to get the abilities

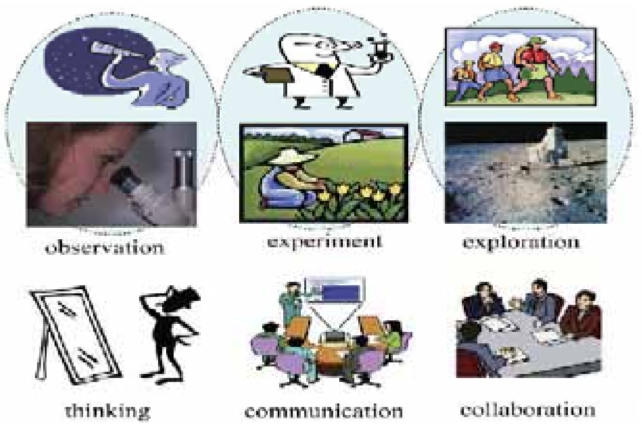


Fig. 3. Abilities of knowledge discovery

of reasoning, inference, optimization, based on the learner's Type-I knowledge. Though these processes are not really of learning, we always demand a fast implementation of problem solving. For this purpose, we need a device that is developed or learned during learning of Type-I knowledge from samples obtained in the same time of problem solving.

In a summary, we have Type-I learning for discovering world knowledge via mining invariant dependence underlying a set of all samples and Type-II learning for problem solving via building up input-response type dependence per sample, as shown in Fig.5.

B. Three ingredients and two challenges

Shown in Fig.6 are three key ingredients of learning. One is gathering a set $\mathcal{X} = \{x_t\}_{t=1}^N$ of samples from the world. The other is finding a learner's architecture with appropriate structures that is able to well accommodate or describe dependence as discussed Fig.5. The third is a learning principle or theory as well as an efficient learning algorithm that implements the learning theory such that dependence underlying the world are learned from the set of samples to the learner's architecture.

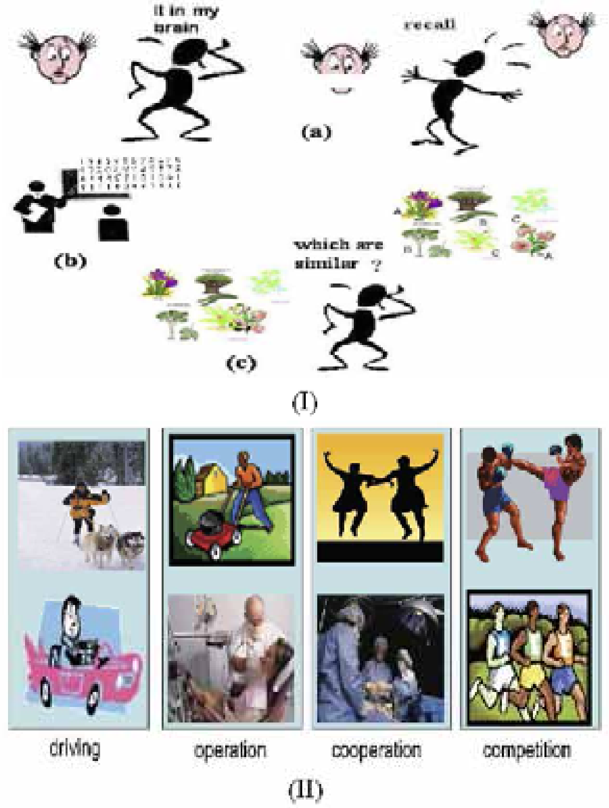


Fig. 4. Skills of problem solving, (I) perception, (II) control.

Intrinsically, learning is a procedure that inevitably bears uncertainty. As illustrated in Fig.7(a), we consider a simple problem of learning a line from samples. Conceptually, a line can be learned usually by two samples. There is uncertainty that we will fail when two samples are same as shown in Fig.7(b). If we get each sample randomly with equal chance from any points on the line, the uncertainty will reduce towards to zero as the number of samples increases. In practice, uncertainty also comes from noises in gathering samples, quantization effects such that a line obtained from two points could be very bad as illustrated in Fig.7(c). Again, as more samples are randomly sampled and each sample comes more equally from any points on the line as illustrated in Fig.7(d), this type of uncertainties will also reduce. Moreover, to find a more complicated structure, another uncertainty comes from not enough number of samples as illustrated in Fig.7(e). Again, we expect that more samples come and each sample comes with equal chance. In a summary, sample gathering is a random sampling process and learning is featured by statistical nature aiming at dependence structures with uncertainties reduced or removed. Thus, we refer it by the term 'statistical learning'.

Strictly speaking, samples can be gathered in an active way or passive way, a learner may particularly seek some

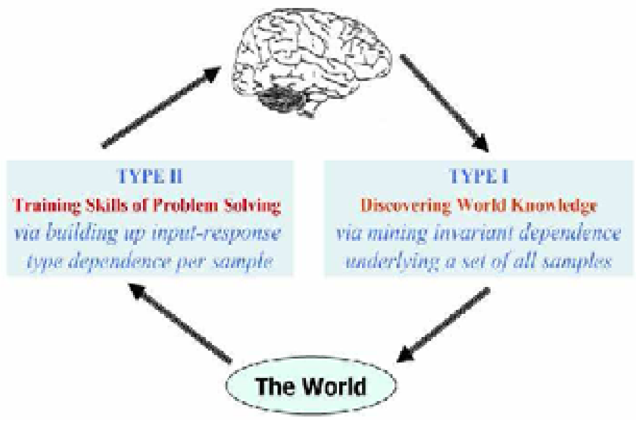

TYPE II
Training Skills of Problem Solving
via building up input-response type dependence per sample
TYPE I
Discovering World Knowledge
via mining invariant dependence underlying a set of all samples
The World

Fig. 5. Two Types of Learning

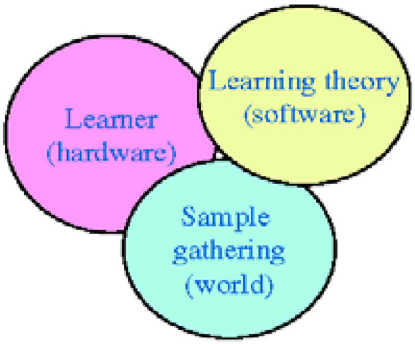


Fig. 6. Key Ingredients of Learning

types of samples according to the learner's prior knowledge or specific attention, and the learning in such a case is sometimes referred as active learning. Though the process of such a learning could be much faster and has certain advantage, it is also easy to bring a biased result. No systematic effort has been made along this direction yet, in spite of some efforts in the literatures. In the fields of statistics, machine learning, neural networks, pattern recognition as well as recently in data mining and bioinformatics, most of past existing studies base on samples that come from the world in a passive way without any learner's interaction, namely, samples are regarded as coming from its world randomly according to its own underlying distribution. We adopt this convention here too.

As to the other two ingredients in Fig.6, we are facing the following two major challenges:

Challenge I: the learner's hardware should be designed not only be able to accommodate but also appropriately match the interested dependence structures underlying the world.

Challenge II: the complexity of the learner's hardware should be appropriately determined to match usually a finite size of samples, namely those reliable dependence structures underlying the samples for representing the underlying world.

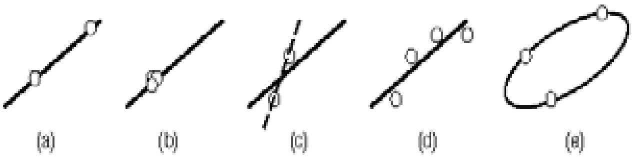


Fig. 7. Uncertainty in samples and random sampling

II. Towards Solving Challenge I

As shown in Fig. 8, the first challenge we encounter is that a learner's architecture should be able to appropriately represent dependence among data.

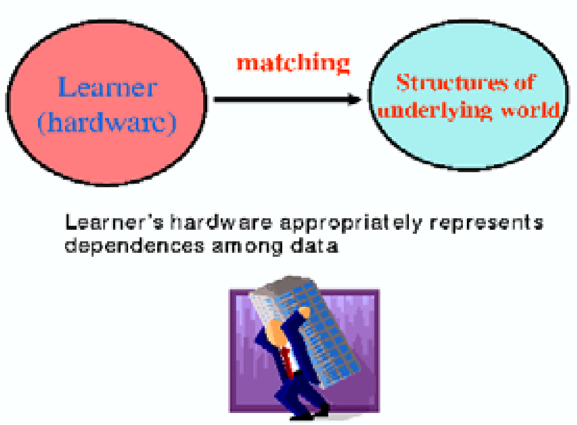

Learner (hardware)
matching
Structures of underlying world
Learner's hardware appropriately represents dependences among data

Fig. 8. Key Challenge I: Learner's hardware appropriately represents dependence among data

A. *Early Efforts: General Purpose vs Specific Purpose*

Started in the statistics literature, one early ambition is to estimate the probabilistic distribution underlying samples since all the dependence structures can be derived from the distribution.

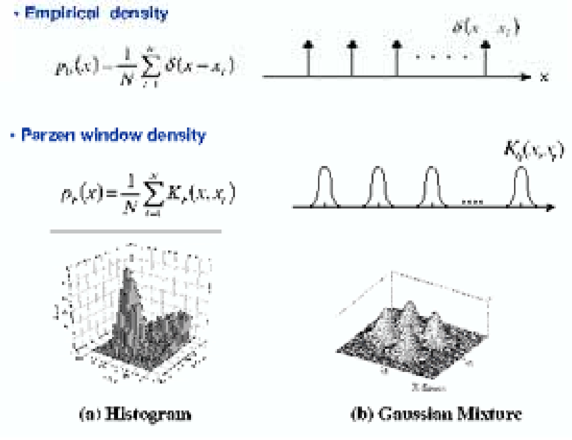
As shown in Fig. 9, the most simple one is directly using and memorizing the entire sample set as a representation of the observed world, which is equivalent to the empirical density:

$$p_0(x) = \tfrac{1}{N}\textstyle\sum_{t=1}^{N}\delta(x-x_t),$$
$$\delta(x) = \begin{cases} \lim_{\delta\to 0}\frac{1}{\delta^d}, & x=0,\\ 0, & x\neq 0, \end{cases} \qquad (1)$$

where d is the dimension of x and $\delta > 0$ is a small number. Improvements have been further proposed by replacing $\delta(x-x_t)$ with a smoothing kernel $K_h(x,x_t)$ as follows:

$$p_h(x) = \tfrac{1}{N}\textstyle\sum_{t=1}^{N}K_h(x,x_t),$$
$$K_h(x,x_t) \text{ is a kernel located at } x_t, \qquad (2)$$

which is usually called a non-parametric Parzen window density estimate [31]. In the simplest case, $K_h(x,x_t)$ is a



• Empirical density

$$p_0(x) = \frac{1}{N}\sum_{t=1}^{N}\delta(x - x_t)$$

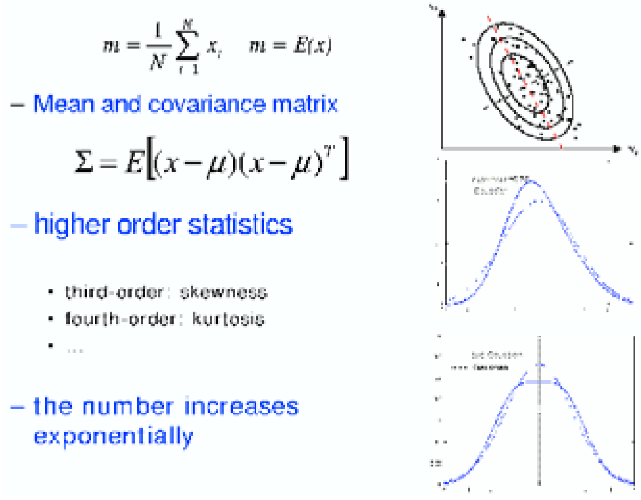
• Parzen window density

$$p_h(x) = \frac{1}{N}\sum_{t=1}^{N}K_h(x, x_t)$$

Fig. 9. General purpose effort: (I) nonparametric density estimation (memorizing samples)

hyper cubic of volume h^d with its center located at x_t, and $p_h(x)$ becomes the widely used histogram estimate as a smoothed version of eq.(1). The smoothness is controlled by a given parameter $h > 0$ that is usually called *smoothing parameter*. The other case is $K_h(x, x_t) = G(x|x_t, h^2I)$, where and hereafter in this paper $G(x|m, \Sigma)$ denotes a Gaussian density with mean vector m and covariance matrix Σ.

Though $p_0(x)$ and $p_h(x)$ tend to the distribution underlying samples as $N \to \infty$, such a nonparametric and nonstructural density estimate works only when the dimension d of x is not high. As d increases, the size N should increase exponentially with d in order to maintain the usefulness of $p_h(x)$. Unfortunately, we often have a finite size N. As a result, the performance deteriorates drastically when the dimension d becomes large, which is usually referred as 'curse of dimensionality'.



$$m = \frac{1}{N}\sum_{t=1}^{N} x_t \quad m = E(x)$$

– Mean and covariance matrix

$$\Sigma = E\left[(x-\mu)(x-\mu)^T\right]$$

– higher order statistics

- third-order: skewness
- fourth-order: kurtosis
- ...

– the number increases exponentially

Fig. 10. General purpose effort: (II) statistics estimation (summarizing ensemble features of samples)

Instead of memorizing each individual sample, as shown in Fig. 10, another early effort of general purpose is extracting features or statistics from an entire ensemble of samples. The most simple and useful one is its mean vector $\mu = Ex$, where and throughout this chapter the notation $E(u) = Eu = E[u]$ denotes the expectation of random variable u. This mean vector μ reflects how samples of x relate each other in a sense of locating around μ, which is also called the first order dependence that describes how each variable $x^{(i)}$ varies depending on its mean $\mu^{(i)} = Ex^{(i)}$. Usually, having only the mean μ is far from enough. We also consider the 2nd order dependencies between every pair of $x^{(1)}, \cdots, x^{(d)}$, i.e., $\sigma_{ij} = E(x^{(i)} - \mu^{(i)})(x^{(j)} - \mu^{(j)}), i = 1, \cdots, d, j = 1, \cdots, d$, which is also written in a matrix $\Sigma = [\sigma_{ij}]$, called the covariance matrix. Actually, it is equivalent to specifying a multivariate Gaussian density with the mean μ and the covariance matrix Σ. Still, having only dependencies up to the 2nd order is not enough for many practical tasks. Naturally, we can further consider dependencies up to any higher order k. However, the number of parameters for representing those dependencies increases exponentially with $O(d^k)$, and thus the size N of samples should increase exponentially with this order to maintain the usefulness of the estimations on these dependencies, which again leads to a problem similar to 'curse of dimensionality'.

Another direction of early efforts is to consider specific density in a parametric form, according to a specific domain knowledge. In the statistics literature, there are tool boxes of specific densities, e.g., a so called exponential family of densities. However, at what situation to choose which specific parametric density needs specific domain knowledge, which are usually not available to us. This nature has made this direction not suitable for a learner who needs a much general learning ability to cover various situations that a learner may encounter.

B. Efforts in Recent Decades: Seeking Dependence Structures

It does not always need and also is not always a good choice to directly seek the probabilistic distribution underlying samples. Not only obtaining the distribution and then getting dependence structures usually incurs expensive computing cost, but also estimating distribution is more vulnerable to a finite size of samples and easy to be lead to poor estimations.

Efforts in recent two decades or more have been made on seeking typical structures that are able to perform major tasks we encounter, especially in the literatures of machine learning and neural networks. Actually, a particular structure implicitly specifies a distribution and may be able to describe a particular class of higher order dependencies.

A number of structures have been studied, but in lack of a systematic view. In the following, we aim at such a purpose from the perspective of two type learning as shown in Fig.5. In general, dependence structures are accommodated in a bi-directional architecture. However, special cases with an one directional architecture are also useful in several learning tasks.

1) Forward Architecture: We start at the special case of *Forward Architecture* that only considers one directional dependence as shown in Fig.11, featured by a mapping $x \to y$ via $y = f(x)$ or $p(y|x)$ that is implicitly described by an appropriate specific structure.

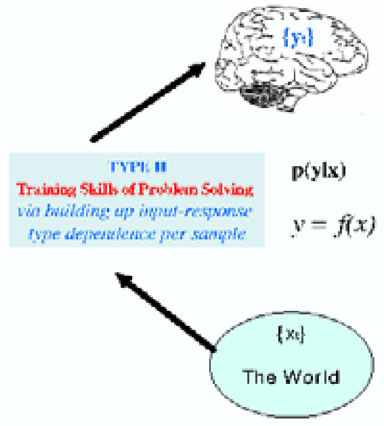

{yt}
TYPE II
Training Skills of Problem Solving
via building up input-response
type dependence per sample
p(y|x)
y = f(x)
{xt}
The World

Fig. 11. Forward Architecture

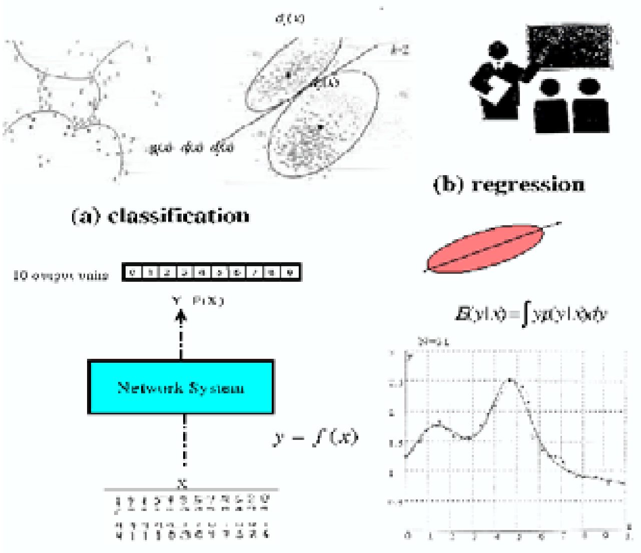


Fig. 12. pair-wise structures

One widely studied type of such dependence structures are *Pair-wise structures*, as shown in Fig.12, which are used to learn a dependence structure from x to y through a set of known sample pairs $T_{x,y} = \{x_t, y_t\}_{t=1}^N$ under the name of supervised learning.

When y is real and $p(y|x)$ is Gaussian, the optimal function of $f(x)$ to fit the set $\{x_t, y_t\}_{t=1}^N$ is the regression $f(x) = E(y|x)$ in a maximum likelihood sense, with various applications such as function fitting, control, prediction, etc. A dependence relation is also considered via intersecting the distribution at a given level for those association rules that are studied widely in the literature of data mining [39], with details referred to Sec.22.2 in [102]. When each element of y takes only 1 or 0, the optimal function of $f(x)$ is one able to classify samples into one of k classes with a minimum classification rate, as encountered in various pattern recognition tasks [30], [33]. In the past two decade, $f(x)$ has been widely implemented by neural networks, e.g., as shown in Fig.13.

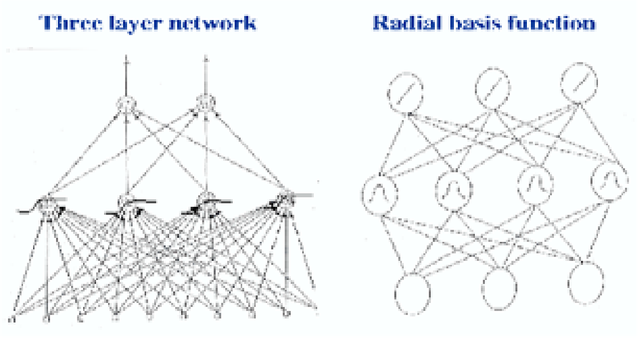


Fig. 13. three layer networks and radial basis function

Another popular type of *Forward Architecture* consists of *transformation structures*, as shown in Fig.14. Instead of knowing a set of sample pairs $\{x_t, y_t\}_{t=1}^N$, we use a parametric structure of $p(y|x)$ to transform observations of x either per each sample into its inner representation y or collectively in term of $p(x)$ by eq.(1) or eq.(2) into

$$p(y) = \int p(y|x)p(x)\mu(dx) \tag{3}$$

so that dependence structure among variables $x^{(1)}, \cdots, x^{(d)}$ are well extracted and represented by $p(y|x)$. From probability theory, if x is mapped by $f(x)$ into a uniform distribution $p(y)$ as shown in Fig.14(c), $f(x)$ is the cumulated distribution function (CDF) of x and thus fully describes dependence structure among variables $x^{(1)}, \cdots, x^{(d)}$.

However, such a $f(x)$ must be a nonlinear map that is difficult to get. Usually, a linear mapping $f(x)$ is considered. Under this linear constraint, the purpose is changed into requiring that y should contain as least as possible redundant information. Thus, a natural choice is $q(y)$ stratifying

$$q(y) = \prod_{j=1}^m q(y^{(j)}). \tag{4}$$

That is, the transform makes x into y with its components being mutually independent. Specially, we are lead either to the well known principal component analysis (PCA) when samples of x come from Gaussian as shown in

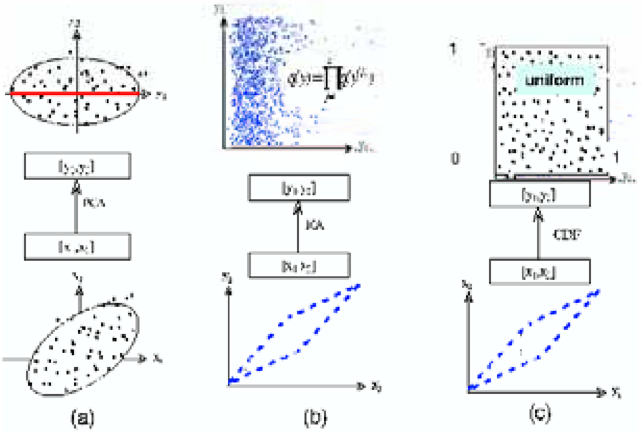


Fig. 14. transformation structure

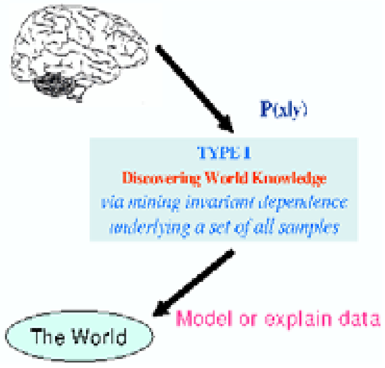
Fig.14(a), or to a popularly studied topic in the recent two decades, namely independent component analysis (ICA), when samples of x come from nonGaussian as shown in Fig.14(b).

2) *Backward Architecture:* The counterpart of *Forward Architecture* is the *Backward Architecture* shown in Fig.15, which describes dependence among variables $x^{(1)}, \cdots, x^{(d)}$ by attempting to reconstruct observations of x from certain inner factors y via $q(x|y)$ in an appropriate parametric form, either collectively via

$$q(x) = \int q(x|y)q(y)\mu(dy) \tag{5}$$

or per each observation x via associating an inner representation y via $q(x|y)q(y)$ and then regarding x as generated from y by a mapping $g: y \to x$ derived from $q(x|y)$.

Conceptually, we can specify $q(x|y)$ and $q(y)$ via using $q(x)$ by eq.(5) to fit samples of x, e.g., in a sense of the maximum likelihood learning, it is computationally difficult to implement since the integral in eq.(5) usually can not be analytically solved.



P(x|y)
TYPE I
Discovering World Knowledge
via mining invariant dependence
underlying a set of all samples
The World
Model or explain data

Fig. 15. Backward Architecture

One implementable category of *Backward Architecture* consists of *Linear Latent structures*, as shown in Fig.16, which are applicable to observations of x that are regarded as generated from

$$x = Ay + \mu + e, \ with \ det[A^T A] \neq 0, \tag{6}$$

where the noise e comes from Gaussian $G(e|0, \Sigma_e)$ and is independent from a inner representation vector y with dimension m, namely, we have

$$q(x|y) = G(x|Ay + \mu, \Sigma_e). \tag{7}$$

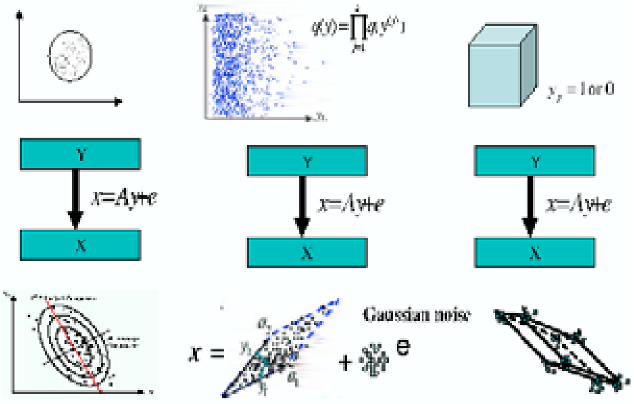


Fig. 16. Linear Latent structures

Similar to the situations in Fig.14, any redundance between any pair $y^{(i)}$ and $y^{(j)}$ should be removed. In a probabilistic sense, it means a distribution $q(y)$ that satisfies eq.(4). Specially, we are lead to the well known factor analysis (FA) for $q(y) = G(y|0, I)$ as shown in Fig.16(a), which has been widely studied and used in the literature of statistics and many other fields since 1956 [6], [69]. In the past decade, studies have also be made for the case that each $y^{(j)}$ comes from a Bernoulli distribution:

$$q(y^{(j)}) = q_j^{y^{(j)}}(1 - q_j)^{1-y^{(j)}}, \tag{8}$$

under the name of binary factor analysis (BFA) or latent trait model as shown in Fig.16(c), which has also been widely studied in several fields [114], [10], [110], [108], [105]. When y is real and $q(y^{(j)})$ is nonGaussian, the integral in eq.(5) still can not be analytically solved. In recent years, such a difficult case has been also studied under the name of independent factor analysis [68], [7], or nonGaussian factor analysis (NFA) [116], [107], with two types of implementing algorithms developed.

One other implementable category of *Backward architecture* consists of *Mixture structures*, as shown in Fig.17, which consider observations of x as coming randomly from each of several distributions, namely a finite mixture as follows [29], [77], [70]

$$q(x) = \sum_{\ell=1}^{k} \alpha_\ell q(x|\theta_\ell). \tag{9}$$

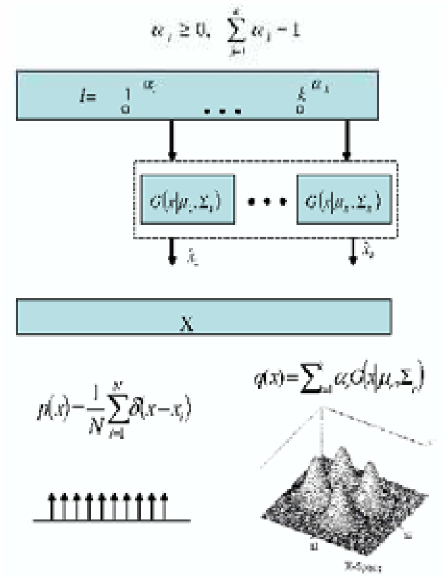

$\alpha_j \ge 0, \ \sum_{j=1}^{k} \alpha_j = 1$
l=
X
$p(x)=\frac{1}{N}\sum_{i=1}^{N}\delta(x-x_i)$
$q(x)=\sum_{l}\alpha_l G(x|\mu_l,\Sigma_l)$

Fig. 17. Mixture structures

One of widely studied case is Gaussian mixture with

$$q(x|\theta_\ell) = G(x|m_\ell, \Sigma_\ell). \tag{10}$$

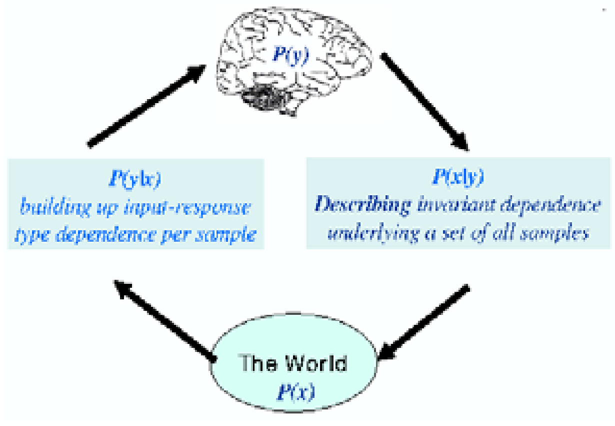

P(y)
P(y|x)
building up input-response
type dependence per sample
P(x|y)
Describing invariant dependence
underlying a set of all samples
The World
P(x)

Fig. 18. bi-directional architectures

3) Bi-directional Architecture: Both the forward architecture and the backward architecture are special cases of a general case shown in Fig.18. For the pairwise structures in Fig.12, we can regard that dependence backwardly from y to x is already given by $T_{x,y}$, while for the transform structures in Fig.14, we can also regard that the dependence backwardly from y to x is also collectively specified by a mapping from the specific $q(y)$ to observations of x. Moreover, the *Backward Architecture* in Fig.15 actually also implies a forward structure as follows

$$p(y|x) = \frac{q(x|y)q(y)}{q(x)} \tag{11}$$

during implementing the maximum likelihood (ML) learning on $q(x)$ by eq.(5). It is also called posteriori distribution in a Bayesian sense.

Structures different from that given by eq.(11) are also used in a bi-directional architecture either under a learning principle that is different from the ML learning or for a purpose of learning regularization, as will be further discussed in Sec.III-E. For examples, given in Fig.19 are some bi-directional versions of the latent structures in Fig.16. Also, given in Fig.20 is an example of bi-directional architecture in Fig.17.

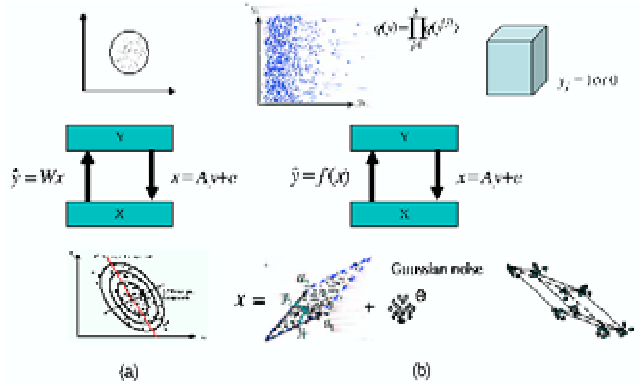


Fig. 19. Bi-directional Latent structures

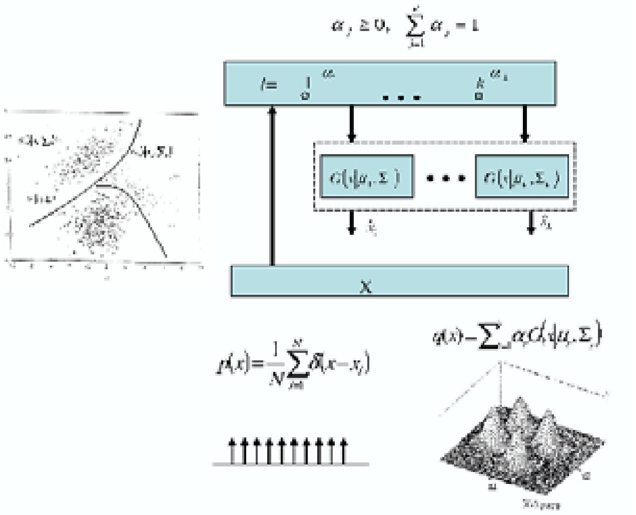

$\alpha_j \ge 0, \ \sum_{j=1}^{k} \alpha_j = 1$
l=
X
$p(x)=\frac{1}{N}\sum_{i=1}^{N}\delta(x-x_i)$
$q(x)=\sum_{l}\alpha_l G(x|\mu_l,\Sigma_l)$

Fig. 20. Gaussian mixture in a Bi-directional architecture

As shown in Fig.21, a motor control task can also be implemented from a bi-directional perspective. The desired position or trace of a movement is indicated by d, the robot arm is intended to approach or follow d, the movement mechanism with a noise disturbance is considered by $q(x - d|y)$ with y being control signal. Moreover, $p(y|x)$ describes how control signal is generated with a noise disturbance in consideration also.

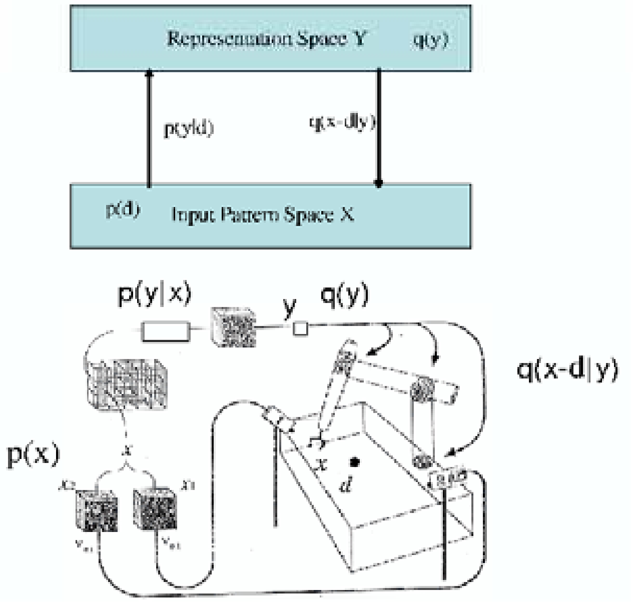


Fig. 21. Motor Control

C. Dependence Structures cross Multi-bodies

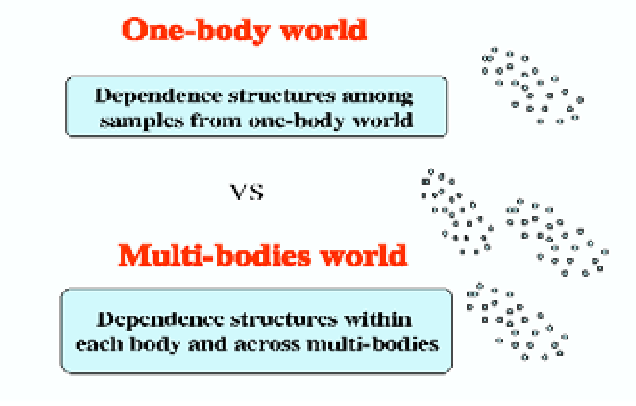

One-body world
Dependence structures among samples from one-body world
VS
Multi-bodies world
Dependence structures within each body and across multi-bodies

Fig. 22. Dependence structures among samples from one body world versus multi-bodies world

Except for the mixture structures in Fig.17, all the pre-discussed structures aim at describing dependence among samples from an one body world or more precisely dependence structures within one body. As illustrated in Fig.25, it is more likely for a learner to encounter a world with many bodies, while it is rare to encounter a pure one-body world. However, studies on dependence structures within an one body world are still useful in a two-fold sense. First, it directly works in the cases that samples from our interested body can be separated from others (easily or in help of some means). Second, its studies provide a foundation to study dependence structures in a world of multi-bodies. Summarized in Fig.23 are typical dependence structures within one-body. The details are referred to [100], [102].

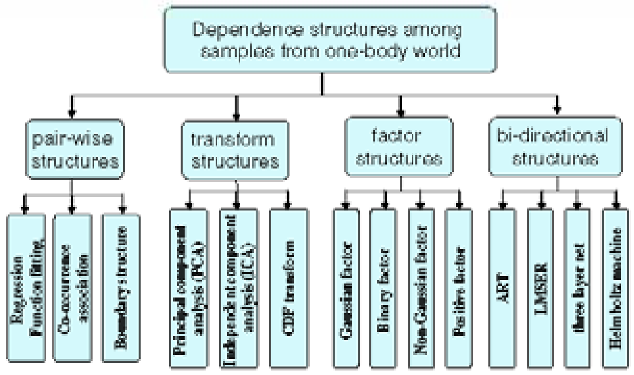

Dependence structures among samples from one-body world
pair-wise structures
transform structures
factor structures
bi-directional structures
Regression Function fitting
Co-occurrence association
Boundary structure
Principal component analysis (PCA)
Independent component analysis (ICA)
CDF transform
Gaussian factor
Binary factor
Non-Gaussian factor
Positive factor
ART
LMSER
three layer net
Helmholtz machine

Fig. 23. Dependence structures among samples from one body world

Illustrated in Fig.24(a), there are multi-bodies A, B, C, D, E, with each body having its own distribution (e.g., as illustrated in the figure for body A). Dependence structures in this multi-body world consists of two parts. One consists of those within each individual body like A, which can be described by those dependence structures in Fig.23. The other consists of dependence structures cross multi-bodies, as illustrated in Fig.24(b). This part can be further divided qualitatively into topology structures and quantitatively into dependence structures among variables across bodies (e.g., dependence between variable x_D from body D and variable x_E from body E). Variety of the two types of dependence structures results in a huge number of possible combinations to study.

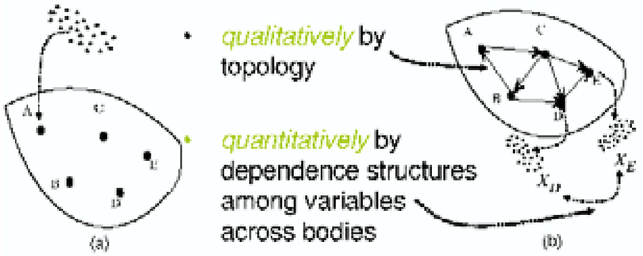

A
C
B
D
E
(a)
qualitatively by topology
quantitatively by dependence structures among variables across bodies
A
C
B
D
E
x_D
x_E
(b)

Fig. 24. Dependence structures among samples from a world of multi-bodies

1) Dependence structures cross multi-bodies with a visible topology: We say that a topology is visible in a sense that not only the topology across multi-bodies is known, but also which sample from which body is known. Shown in Fig.25(a)&(b), one simple but widely studied case is that multi-bodies are ordered on a directional line segment and body t is observed via a variable x_t. Thus, we have series $x_t, x_{t-1}, \cdots, x_{t-p}$, which is usually called a waveform or time series since one typical example of this line topology is time. In this case, we need to consider not only dependence structures among elements of x_t but also across times

$x_t, x_{t-1}, \cdots, x_{t-p}$. Generally, it should be represented by a stochastic process $p(x_t, x_{t-1}, \cdots, x_{t-p})$, which again is difficult to estimate directly via empirical distribution.

The pair-wise structure in Fig.12 becomes now a temporal regression $y_t = f(x_t, x_{t-1}, \cdots, x_{t-p})$. The simplest one is what is called linear regression

$$y_t = \sum_{i=0}^{p} a_i x_{t-i} + \varepsilon_t, \tag{12}$$

where ε_t is a white noise with $E\varepsilon_t = 0$ and

$$E\varepsilon_t\varepsilon_\tau^T = \gamma\bar{\delta}(t-\tau), \gamma > 0,\ \bar{\delta}(u) = \begin{cases} 1, & u = 0, \\ 0, & u \neq 0. \end{cases} \tag{13}$$

The transformation structure in Fig.14 applies directly too, specially when y_t is a vector.

When $y_t = x_{t+1}$, another typical example is the following autoregressive moving-average (ARMA) [55]

$$x_{t+1} = \sum_{i=0}^{p} a_i x_{t-i} + \varepsilon_t + \sum_{j=1}^{q} b_j \varepsilon_{t-j}. \tag{14}$$

which degenerates to the autoregressive (AR) model when $q = 0$. Further considering the time-varying variance of ε_t, we can also get ARCH, GARCH [15].

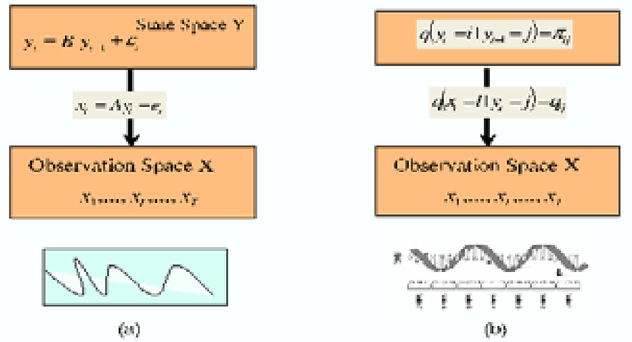

State Space Y
$y_t = B y_{t-1} + \varepsilon_t$
$x_t = A y_t + e_t$
Observation Space X
$x_1, \dots, x_t, \dots, x_T$
$q(y_t = i \mid y_{t-1} = j) = \pi_{ij}$
$q(x_t = l \mid y_t = j) = q_{lj}$
Observation Space X
$x_1, \dots, x_t, \dots, x_T$
(a)
(b)

Fig. 25. State space model and Hidden Markov model

Similar to the linear latent structures in Fig.16, temporal relation can also be described by a backward architecture in a structure as shown in Fig.25.

Specifically, when x_t is real, as shown in Fig.25(a), we have the well known state space model

$$x_t = Ay_t + \mu + e_t,\ y_t = By_{t-1} + \varepsilon_t, \tag{15}$$

which has been widely studied in the literature of control theory. Provided that A, B, μ and the variances of e_t and ε_t are known, the task is to estimate y_t, which is made by eq.(11) that is equivalent to the well known Kalman filter [51]. Several years ago [113], [110], the state space model is re-visited as an temporal extension of eq.(6) under the name of temporal factor analysis (TFA), with the components of y_t required to be uncorrelated while A, B, the variances of e_t and ε_t are relaxed to be unknown.

When x_t is discrete, as shown in Fig.25(b), we have the well known hidden Markov models (HMM) and extensions, which have been widely used in speech processing and bioinformatics [76], [8]. Details are referred to [100]. Moreover, shown in Fig.25(c) and Fig.27 are examples with visible topology of image, tree and graph.

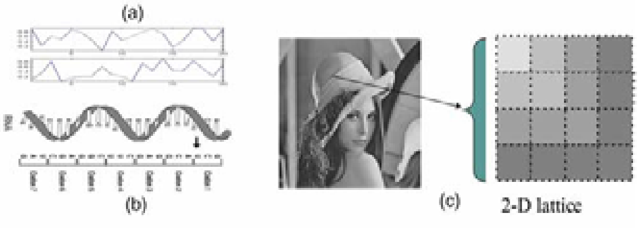


Fig. 26. Lattice topology

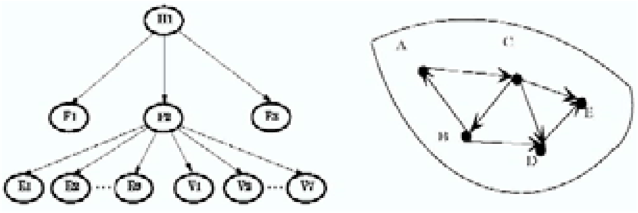


Fig. 27. Tree and graph topology

2) *Dependence structures cross multi-bodies with a invisible topology:* When the topology across multi-bodies is invisible and which sample from which body is unknown, we need both to recover the topology and to classify samples to its corresponding body. However, it is very difficult to learn topology from samples. Currently, only two special cases have been studied.

One widely studied case is *Mixture structures* by eq.(9), where we only consider *null topology*. That is, among the information across multi-bodies, we ignore the topology across multi-bodies. If we already knew the ID information of each sample, i.e., which sample from which body, the problem degenerates to a problem of mining dependence structure among each one body world. However, in many applications, we do not known but need to recover the ID information.

Each body can be either a Gaussian as shown in Fig.17, or a body with other structures as shown in Fig.28. Since each body locates at a different site, we also call it local dependence structure.

The other widely studied case is the well known topological map that is able to demonstrate an important topology type, which is induced from concepts such as 'similar' and 'near', etc, via a spatial relation among bodies located in the Euclidean space. Considering a regular d-dimensional lattice topology, we attempt to locate each body on one node of the lattice such that objects locating topologically

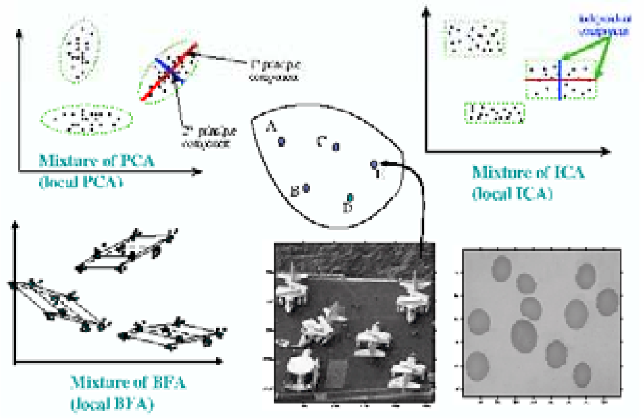


Fig. 28. Typical local dependence structures

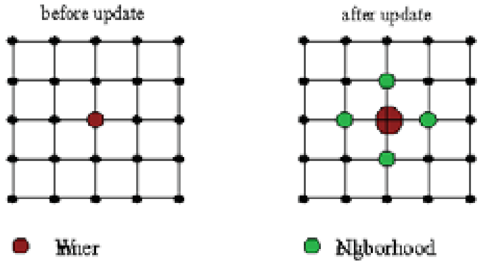


Fig. 29. One member wins, a family gains

near should be similar to each other. A direct placement of all the objects on such a lattice, in help of a given similarity measure to judge whether two objects are similar, is computationally a hard combinatorial problem. Interestingly, this problem has been heuristically implemented approximately in help of a biological brain dynamics of self-organization [67], featured by a Mexican hat type interaction, namely, neurons in near neighborhood excite each other with learning, while neurons far away inhibit each other with de-learning.

Computationally, such a dynamic process can be further simplified by certain heuristic strategies. One widely used is the well known Kohonen self-organizing map [58] that implements a strategy of *one member wins, a family gains* That is, as long as one member wins in the winner-take-all competition, all the members of a family will gain regardless whether other members are strong or not. As shown in Fig.29, with each node on the lattice that represents an body or class, a winner-take-all competition is made per sample to get the winner

In [108], we also get an alternative strategy of *strongers gain and then teaming together*. That is, a number of strongers in competition will be picked as winners who not only gain learning but also are teamed together to become neighbors. As experimentally demonstrated in [22], this strategy can speed up self-organization, especially at the early stage of learning. Also, we can use it at an early stage and subsequently switching to the Kohonen map.

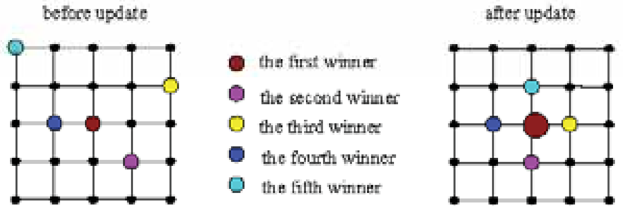


Fig. 30. Strongers gain and then teaming together

Summarized in Fig.23 are typical dependence structures within one-body. The details are referred to [100], [102].

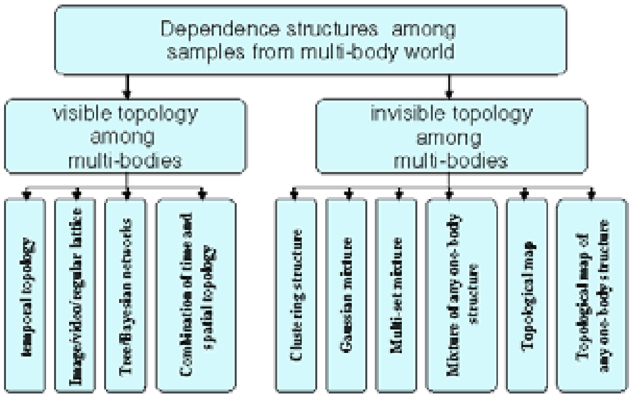

Dependence structures among samples from multi-body world
visible topology among multi-bodies
invisible topology among multi-bodies
temporal topology
Image/video/regular lattice
Tree/Bayesian networks
Combination of time and spatial topology
Clustering structure
Gaussian mixture
Multi-set mixture
Mixture of any one-body structure
Topological map
Topological map of any one-body structure

Fig. 31. Dependence structures among samples from multi-body world

D. Bayesian Ying-Yang system as a general framework

In fact, all the previous introductions on typical dependence structures have been made on a general framework that emphasizes to jointly consider a forward architecture and a backward architecture, with the common inner representations subject to $q(y)$ as a bridge. That is, these typical dependence structures can also be summarized as special cases of the bi-directional architecture shown in Fig.18.

In the general form, as shown in Fig.32, we actually consider the joint distribution of the observation world and its inner representation via the following the two types of Bayesian decomposition:

$$p(x, y) = p(y|x)p(x), \;\; q(x, y) = q(x|y)q(y), \qquad (16)$$

In a compliment to the famous Chinese ancient Ying-Yang philosophy, the decomposition of $p(x, y)$ coincides the Yang concept with a visible domain from $p(x)$ regarded as a Yang space and the forward pathway by $p(y|x)$ as a Yang pathway. Thus, $p(x, y)$ is called Yang machine. Similarly, $q(x, y)$ is called Ying machine with an invisible domain from $q(y)$ regarded as a Ying space and the backward

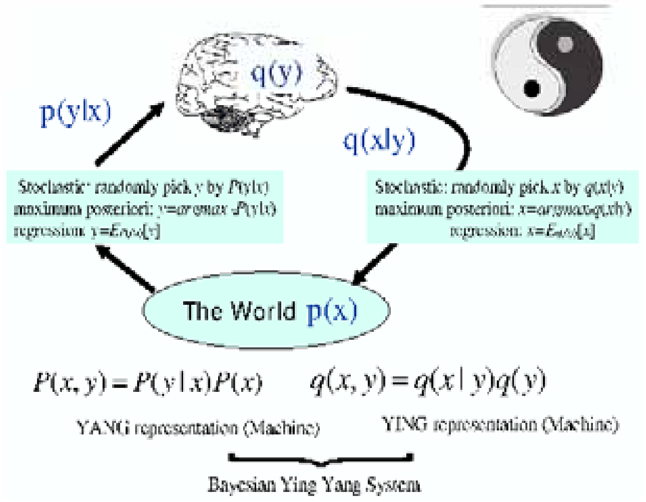

q(y)
p(y|x)
q(x|y)
Stochastic: randomly pick y by P(y|x)
maximum posteriori: y=argmax P(y|x)
regression: y=E_{P(y|x)}[y]
Stochastic: randomly pick x by q(x|y)
maximum posteriori: x=argmax q(x|y)
regression: x=E_{q(x|y)}[x]
The World p(x)
P(x, y) = P(y | x)P(x)
q(x, y) = q(x | y)q(y)
YANG representation (Machine)
YING representation (Machine)
Bayesian Ying Yang System

Fig. 32. A general framework for learner's architecture

pathway by $q(x|y)$ as a Ying pathway. Such a pair of Ying-Yang machines is called *Bayesian Ying-Yang (BYY) system*.

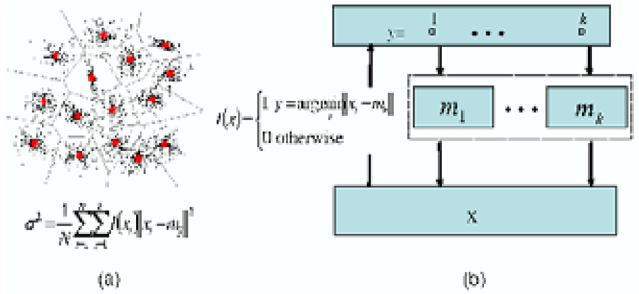

y=
1
k
m_1
m_k
X
I(x)= 1 y = argmin ||x_t − m_j||; 0 otherwise
σ² = 1/N ΣΣ I(x_t) ||x_t − m_j||²
(a)
(b)

Fig. 33. Least mean square error clustering

In the idealistic case, the joint density of x, y should be the same regardless in what kind of representation, i.e.,

$$p(x, y) = p(y|x)p(x) = q(x, y) = q(x|y)q(y), \tag{17}$$

which can be regarded as an extension of the concept of deterministic inverse function. Two functions $x = g(y), y = f(x)$ are said to be mutually the inverse of the other if we have $x = g(f(x))$ for all $x \in D_x$ and $y = f(g(y))$ for all $y \in D_y$, where D_x and D_y are correspondingly the domains of x, y, respectively. On a set of samples of x_t, $x = g(y)$ and $y = f(x)$ are mutually inverse to each other is equivalent to the case that eq.(17) holds with $p(x)$ by eq.(1), $q(x|y) = \delta(x - g(y))$, $p(y|x) = \delta(y - f(x))$, and $q(y) = p(x)/|f'(x)|, x = g(y)$. In other words, the equality by eq.(17) includes the conventional concept of functions inverse as a special case.

It should be noticed that the inverse concept by eq.(17) is usually very different from the conventional one. It is not simply a mutually inverse relation between $p(y|x)$ and $q(x|y)$. As illustrated in Fig.33, in a conventional least mean square error (MSE) clustering, the dependence structure for $q(x|y)$ consists of mean vectors $m_1, \cdots, m_k$ from which we can obtain the boundary line segments in Fig.33(a), i.e., the dependence structure for $p(y|x)$. In a contrast, we are not able to determine the mean vectors $m_1, \cdots, m_k$ from knowing the boundary line segments. However, we will become able to estimate the mean vectors $m_1, \cdots, m_k$ from the boundary line segments when we also know $p(x)$ by eq.(1) or equivalently a set of samples of x. In other words, the inverse concept should also involve $p(x)$.

Actually, the conventional inverse concept $x = g(f(x))$ is required to hold for every $x \in D_x$ uniformly, which is an extreme case of $p(x)$. In a probabilistic sense, $p(x)$ by eq.(1) bears this uniform nature. Thus, the conventional inverse concept and the inverse concept by eq.(17) meet in this case, as above discussed. Beyond this, the equality by eq.(17) also holds in other cases, e.g., with $p(x)$ by eq.(1), the inverse concept by eq.(17) holds for any given $p(y|x)$ as long as $q(y) = \int p(y|x)p(x)dx$ and $q(x|y) = [p(y|x)/q(y)]p(x)$, and $q(y) = p(x)/|f'(x)|, x = g(y)$.

This inverse concept may also be applied to the cases that it does not hold directly on a set of samples of x, with a given parametric $q(x|y)$ and $p(x)$ by eq.(1). In these cases, we use $q(x)$ by eq.(5) to fit a set of samples of x (or equivalently eq.(1)), and then we indirectly use the obtained $q(x)$ as $p(x)$ in eq.(17), the equality will still hold when $p(y|x)$ is given by eq.(11). In other words, this $p(y|x)$ and $q(x)$ by eq.(5) is an inverse of $q(x|y)$ and $q(y)$ in a Bayesian sense.

However, the equality by eq.(17) generally does not hold, because $p(x)$ by eq.(1) or eq.(2) bases on only a finite size of samples, and $p(y|x)$, $q(x|y)$ and $q(y)$ are subject to this or that kind of constraints. In practice, $p(y|x)$ and $q(x|y)$ can become mutually inverse of each other only approximately in a Bayesian sense, or in an other sense.

The above discussed coupling between the Ying-Yang machines not only further supports but also depicts our belief that two types of intelligent abilities in Fig.1 and their dependence structures, as well as the corresponding two types of learning in Fig.5, are closely coupled together, and thus should be modelled and investigated jointly and systematically.

With new insights and new results, a number of widely studied dependance structures in literature can be summarized via special cases of the three architectures of BYY system, as shown in Fig.34. Readers are further referred to Sec.22.4 and Sec.22.5 of [102], where a much more systematic view is given on various dependence structures.

In fact, simultaneously building up models for both the pathways $x \to y$ and $y \to x$ has been widely adopted as a fundamental sprit in various studies of brain theory and neural networks. Typical efforts include Carpenter & Grossberg's ART theory [20] and Hinton and colleagues'

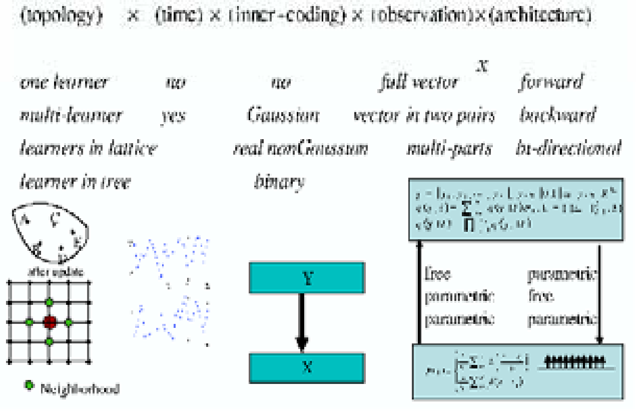


Fig. 34. Integrated structures

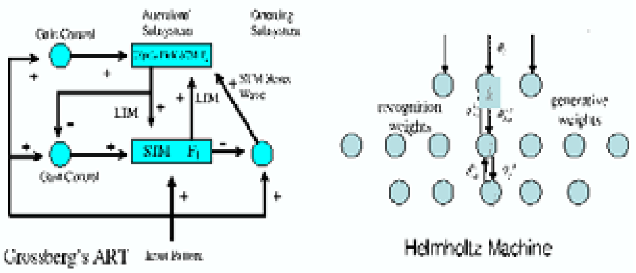


Fig. 35. bi-directional structures

Helmholtz machines and wake-sleep learning [40], [28], as shown in Fig.35, as well as Kawato's theory on cerebellum and motor control [56]. Moreover, the LMSER self-organizing rule proposed in 1991 [139] is also such an effort. The basic sprit of the LMSER self-organization has been further developed into the above discussed BYY system in Fig.32 as a general statistical learning perspective for systematically understanding learning tasks of various dependence structures with new insights, which is firstly proposed in 1995 [130] and then systematically developed in past years. Readers are referred to recent systematical summaries in [108], [109], [105], [106], [107], [100], [101], [102], [103]. Beyond providing a unified perspective, these studies also lead to a new theory under the name of BYY harmony learning for tacking the second challenge of statistical learning.

III. Towards Solving Challenge II

A. *Large number law, parameter learning, and model selection*

As shown in Fig. 9, the density estimation by eq.(1) can be regarded as memorizing N samples, with take pieces of evidence per each piece by $1/N$. It is supported by the large number law firstly obtained by Kolmogorov and Smirnov in 1930's. That is, the error of using the density estimation by eq.(1) as the true density, where the samples come, will tend to zero as the sample size $N \to \infty$, in a sense shown in Fig. 36.

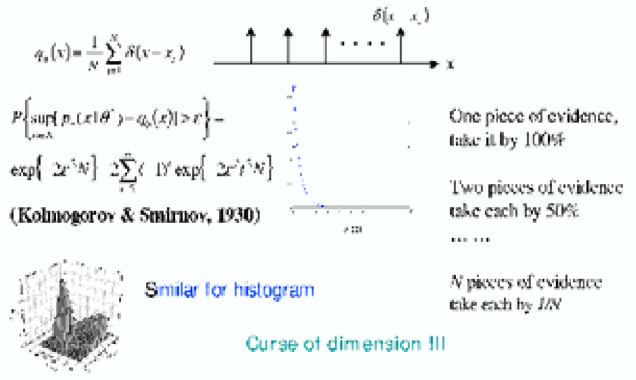

$\delta(x-x_t)$
$q_h(x) = \frac{1}{N}\sum_{t=1}^{N}\delta(x-x_t)$
x
$P\left\{\sup_{x} |p_*(x|\theta^*) - q_h(x)| > \varepsilon\right\} \sim$
$\exp\{-2\varepsilon^2 N\}$
(Kolmogorov & Smirnov, 1930)
One piece of evidence, take it by 100%
Two pieces of evidence take each by 50%
…… …
N pieces of evidence take each by 1/N
Similar for histogram
Curse of dimension !!!

Fig. 36. The large number law: nonparametric case

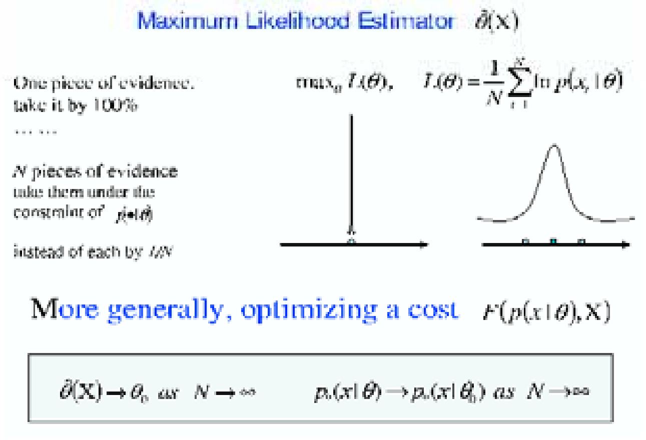
Directly given a parametric form $p(x|\theta)$ or indirectly via a backward structure by eq.(5), as shown in Fig. 37, estimating θ by the maximum likelihood learning can be regarded as taking N pieces of evidence under the constraint of $p(\cdot|\theta)$. It can be also regarded as using a template $p(\cdot|\theta)$ to match a set X of N samples. The estimate $\hat{\theta}(X)$ obtained in such a way will tend to the true value θ_0 as $N \to 0$, if the template happens to be the true density form $p_*(\cdot|\theta)$, according to the large number law obtained during the 50's-60's of the past century.

For the forward architecture in Fig. 11, the above statements on the ML learning applies directly to the pairwise structures in Fig. 12. While for the transformation structures in Fig.14, it can be regarded as using $p(y)$ by eq.(3) to match a desired target that satisfies eq.(4), instead of matching between template and the sample X. Generally speaking, both types of matching can be regard as using a template to match the sample set X to minimize a cost $F(\theta, X)$, from which we can get the large number law similarly.

However, there is no an oracle who tells us the true structure of form $p_*(\cdot|\theta)$. To avoid this difficulty, we consider a family $\mathcal{F}$ of density function forms $p(x|\theta_j)$, $j = 1, \cdots, k, \cdots$ with each sharing a same configuration but its structural scale increasing with k such that $\mathcal{P}_1 \subset \mathcal{P}_2 \subset \cdots \mathcal{P}_k \subset \cdots$, where $\mathcal{P}_j = \{p(x|\theta_j)|\forall\theta_j \in \Theta_j\}$, with three examples given in Fig.38. Provide that there is a $k*$ and $\theta(k^*)$ such that $p(x|\theta*(k))$ is equal or close to the true distribution $p_*(x|\theta_0)$. As a result, we are facing two tasks: estimating $\theta(k^*)$ and to select an appropriate structural scale $k*$. The former is called *parameter learning*, and the latter is called *model selection*.

B. *Challenge II, existing approaches, and two step implementation*

We are not able to collect enough samples either because not a plenty of resources or because not an enough speed



Maximum Likelihood Estimator $\hat{\theta}(X)$

One piece of evidence, take it by 100%

… …

N pieces of evidence take them under the constraint of $p(\bullet|\hat{\theta})$

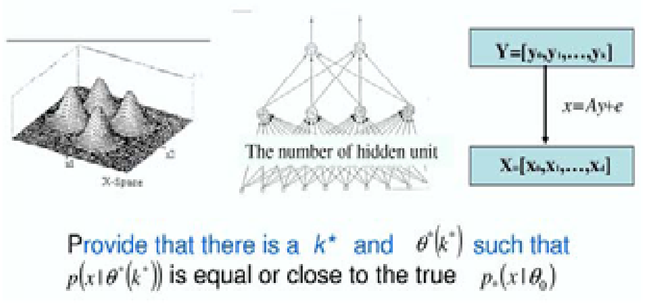
instead of each by $1/N$

$\max_\theta L(\theta)$, $L(\theta)=\frac{1}{N}\sum_{t=1}^{N}\ln p(x_t|\theta)$

More generally, optimizing a cost $F(p(x|\theta),X)$

$\hat{\theta}(X)\to\theta_o$ as $N\to\infty$ $\quad p_o(x|\hat{\theta})\to p_o(x|\theta_0)$ as $N\to\infty$

Fig. 37. The large number law: parametric case



Provide that there is a k^* and $\theta^*(k^*)$ such that $p(x|\theta^*(k^*))$ is equal or close to the true $p_o(x|\theta_0)$

Fig. 38. Two tasks of learning : model selection and parameter learning

to catch the dynamic changing of world. That is, what we usually encounter is a finite number N of samples in $\mathcal{X}$ and thus the large number law does not apply. Even badly, as shown in Fig. 39(a), the error or a cost $F(\theta, X)$ of fitting a finite size of samples decreases monotonically as k increases. That is, it looks that larger the k is, the better it is.

Unfortunately, it is not true. As shown in Fig. 39(a), there are one curve for the fitting error and the other curve for the generalization error of its performances on new samples that have not been used for learning but from a same underlying true distribution. The two curves have a same tendency as k increase at the beginning, which means that the mist-fitting error reduces as the learner's structural scale increases from a small one. At the one point k^*, the generalization error reaches its minimum. As $k > k^*$ further increases, the generalization error increases while the fitting error still reduces. In other words, we are unable to select an appropriate k^* according the fitting error that we can observe. This can also be observed in Fig. 39(b) where the fitting error becomes zero when $k = 8$ with each sample as a cluster, while it is obvious that there are only two clusters in the figure.

This observation may also be illustrated in help of Fig. 39(b). Given a finite size of training samples, a model is illustrated by an ellipse with the scale of model indicated by the elliptic area. The number of training samples that remain uncovered by a model indicates the fitting error, while the area occupied by neither samples or the black shadow indicates where the model can generalize but disagree with the true model, i.e., the generalization error. We can observe that the fitting error reduces as the area of a model increases, but after being large enough to cover the true model, the generalization error increases as the area further increases. Similar observation can also be obtained from the function fitting problem in Fig. 39(d).

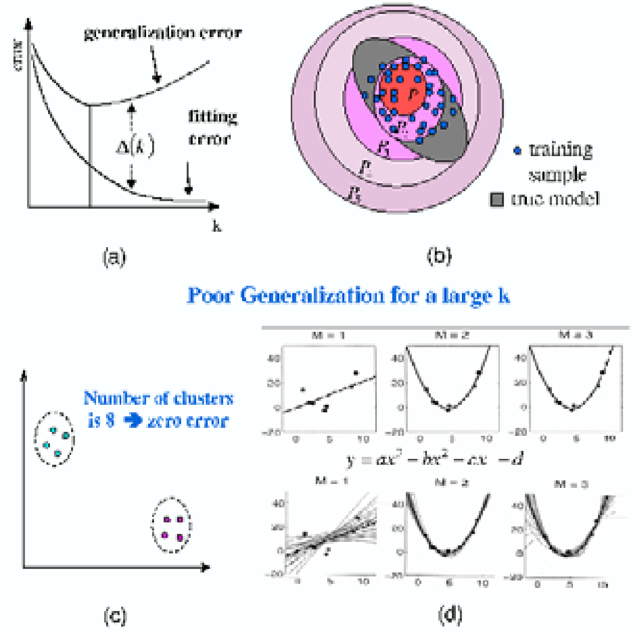


Fig. 39. A hard problem : how to get the difference $\Delta(k)$ between the error of fitting a finite size N of samples and the error on new samples from a same underlying structure

As shown in Fig. 40, we have to face another challenge, that is, how to find an appropriate scale k, namely, how to let the complexity of the learner's hardware to be appropriately determined to match a finite size of samples. In other words, how to find a reliable dependence structure underlying the samples for representing the underlying world.

The challenge is actually very hard to tackle by its nature. This difficulty can be observed from Fig.39. The key point is directly or indirectly estimate difference $\Delta(k)$. Though the fitting error is already available during making learning on a given training set of samples, it needs a testing set of new samples to evaluate the learned model to get its generalization error. The reality is that there is only a finite or small size of samples, which often is already not enough even all of the samples are used for training. Such dilemma makes it doomed that a precise estimation of generalization

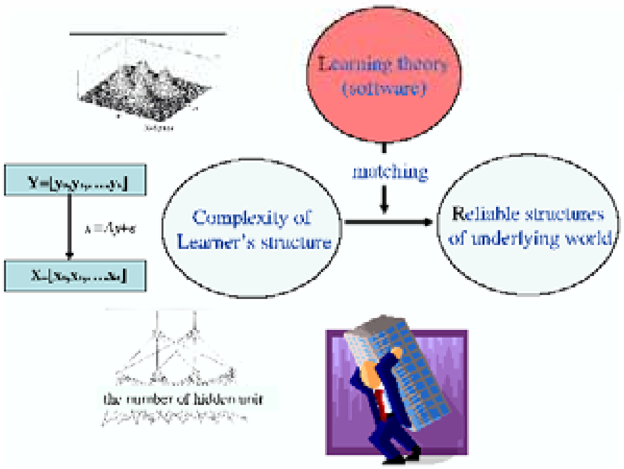


Fig. 40. Key Challenge II: Complexity of Learner's structure matching the size of samples

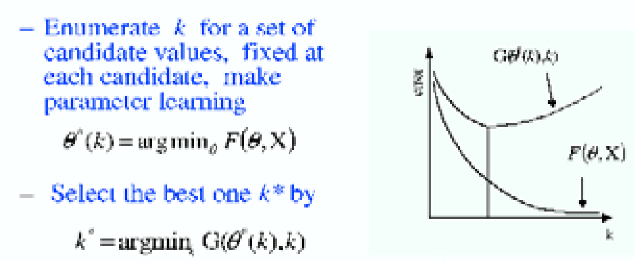
error or thus $\Delta(k)$ is impossible. What we can do is to seek a rough estimation or a bound as good as we are able to.

In the past 30 or 40 years, many efforts have been made towards this challenge, both in the literature of statistics under the name of model selection and in the literature of machine learning under the name of learning theory. Typical results can be roughly summarized into the following typical streams:

- *Towards estimating generalization error by experiments* Studies of this type are mostly made under the name of cross-validation (CV) [86], [87], [88], [81], 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 and a different testing set.
- *Towards estimating bounds of general error via theoretical analysis* A bound of $\Delta(k)$ is estimated in help of analyzing the relation between generalization error and structural complexity of the model. One popular example is the VC dimension based learning theory [97].
- *Towards minimizing information divergence* The discrepancy between the true model and the estimated model is minimized in help of Kullback-Leiber information. Examples include Akaike information criterion (AIC) [1], [2], [3] as well as its extensions AICB, CAIC, etc, [89], [17], [18], [47], [21].
- *Towards Optimizing Bayesian inference* Seeking an optimal Bayesian posteriori inference in help of introducing a prior and estimating the marginal distribution. The typical example is Bayesian inference criterion (BIC) [85], [55], [71] and equivalently those studies under the name of Minimum Description Length (MDL) [79], [80], [65], [66], [25], [41].
- *Towards the Ockham's principle* 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 represent. Typical examples include those studies under the name of minimum message length (MML) theory [93], [94], [95] and the name of Minimum Description Length (MDL) [79], [80], [65], [66], [25], [41].

Conceptually, the first three all focus on the discrepancy between the true model and the estimated model. The first two both target at generalization error, while the third one uses information instead of generalization error to measure the discrepancy. Interestingly, the last two are conceptually different but actually lead to a same criterion after certain mathematical deviation, which hints that optimal Bayesian inference is reached by a parsimonious model. Further detailed discussions on the above approaches are referred to Sec. 23.2 in [103].

As shown in Fig.41, all the above approaches have to be implemented in a two-stage way, that is very computational extensive in implementation, since it needs to repeat parameter learning $\theta^*(k) = arg\min_\theta \ F(\theta, X)$ for a large number of times before becoming able to start selection on one appropriate k under a criterion $G(\theta^*(k), k)$ obtained by anyone of the above approaches. Thus, it is difficult to apply these approaches to many practical uses, especially in real time problems.



– Enumerate k for a set of candidate values, fixed at each candidate, make parameter learning

$$\theta^*(k) = \arg\min_\theta F(\theta, X)$$

– Select the best one k^* by

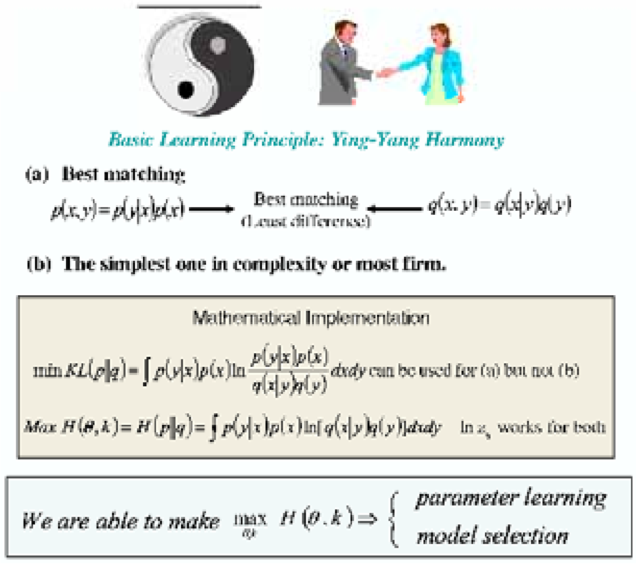
$$k^* = \arg\min_k G(\theta^*(k), k)$$

Fig. 41. Two step implementation

C. BYY harmony learning: (I) automatic model selection

For those dependence structures that have been summarized in Sec.II-D, as well as any new dependence structure that can be summarized in this way, we can tackle the Challenge II by a new theory called *Ying Yang harmony* via the Bayesian Ying-Yang system as shown in Fig.32. More precisely, the Ying-Yang pair by eq.(16) is learned coordinately such that the pair is matched in a compact way as the Ying-Yang sign shown in Fig.42. In other words, learning is made in a twofold sense that

- *Best matching* the difference between the two Bayesian representations in eq.(16) should be minimized.
- *Least complexity* the resulted entire Bayesian Ying-Yang (BYY) system should be of the least complexity.


Basic Learning Principle: Ying-Yang Harmony

(a) Best matching

$p(x,y)=p(y|x)p(x) \longrightarrow$ Best matching (Least difference) $\longleftarrow q(x,y)=q(x|y)q(y)$

(b) The simplest one in complexity or most firm.

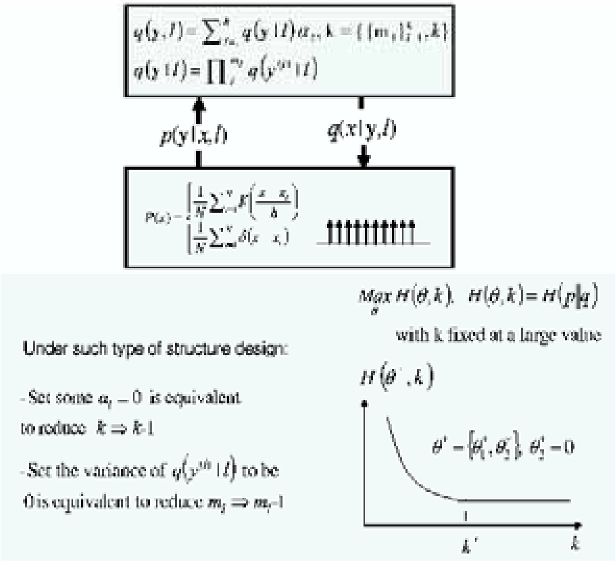
Mathematical Implementation

$\min KL(p\|q)=\int p(y|x)p(x)\ln\frac{p(y|x)p(x)}{q(x|y)q(y)}dxdy$ can be used for (a) but not (b)

$Max\ H(\theta,k)=H(p\|q)=\int p(y|x)p(x)\ln[q(x|y)q(y)]dxdy \quad \ln z_q$ works for both

We are able to make $\max_{\theta k} H(\theta,k) \Rightarrow \begin{cases} \text{parameter learning} \\ \text{model selection} \end{cases}$

Fig. 42. Bayesian Ying-Yang (BYY) Harmony Learning


$q(y,l)=\sum_{l=1}^{k} q(y|l)\alpha_l, \mathbf{k}=\{\{m_l\}_{l=1}^{k},k\}$
$q(y|l)=\prod_j^{m_l} q(y^{(j)}|l)$
$p(y|x,l)$
$q(x|y,l)$
$p(x)=\begin{cases}\frac{1}{N}\sum_{t=1}^{N}K\left(\frac{x-x_t}{h}\right)\\ \frac{1}{N}\sum_{t=1}^{N}\delta(x-x_t)\end{cases}$
$H(\theta^*,k)$
$\theta^*=\{\theta_1^*,\theta_2^*\}, \theta_2^*=0$
k^*
k

$Max_{\theta}\ H(\theta,k),\ H(\theta,k)=H(p\|q)$

with k fixed at a large value

Under such type of structure design:

- Set some $\alpha_l=0$ is equivalent to reduce $k \Rightarrow k$-1

- Set the variance of $q(y^{(j)}|l)$ to be 0 is equivalent to reduce $m_l \Rightarrow m_l$-1

Fig. 43. Parameter Learning with Automated Model Selection

We call it *Bayesian Ying-Yang Harmony Learning* in consideration of *Ying Yang harmony* within Bayesian Ying-Yang system.

Mathematically, both *best matching* and *least complexity* can be realized by implementing [130], [110], [108]

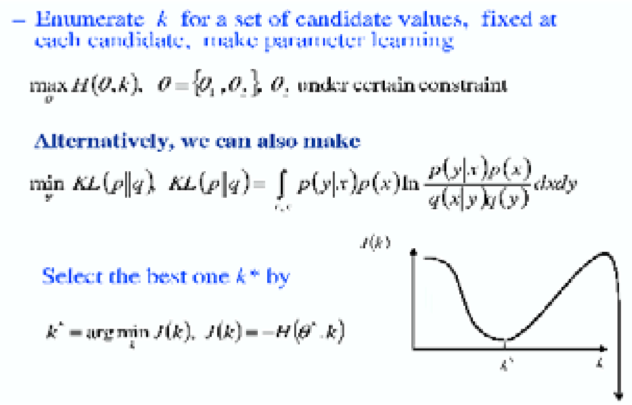
$$\max_{\theta,\mathbf{k}} H(p\|q), \ H(p\|q) = \int p(y|x)p(x)\ln[q(x|y)q(y)]\mu(dx)\mu(dy) - \ln z_q, \tag{18}$$

where θ consists of all the unknown parameters in $p(y|x)$, $q(x|y)$, and $q(y)$ as well as $p(x)$ (if any), while $\mathbf{k}$ is a set of scale parameters of the inner representation y. Specifically, $\mathbf{k}$ consists of only k for the number of bodies in a mixture structure in Fig.17, and $\mathbf{k}$ consists of only the dimension m of y in a linear latent structure in Fig.16. While for a local dependence structure in Fig.15, $\mathbf{k}$ consists of not only k but also either m when y has a same dimension everywhere or $\{m_\ell\}$ when y has a different dimension locally in each body, as shown in Fig.43.

The task of determining θ is called *parameter learning*, and the task of selecting $\mathbf{k}$ is called *model selection* since a collection of specific BYY systems by eq.(16) with different scale values corresponds to a family of specific models that share a same system configuration but in different scales. In addition, the term z_q imposes regularization on learning, which will be further discussed in Sec.III-E. In the simplest case, we can set $z_q = 1$ without regularization.

For those previously discussed learning theories, parameter learning should be firstly made under a matching or fitting principle (especially maximum likelihood principle), which is different from the principle underlying each of these learning theories. A salient difference of BYY harmony learning is that both *parameter learning* and *model selection* are made underlying the same harmony principle. This salient feature makes it possible that model selection can be made automatically during parameter learning.

More precisely, for the inner representation in Fig.43, best harmony will drive α_ℓ to 0 if it is extra, which is equivalent to reducing k into $k-1$. Also, best harmony will drive the variance of $y^{(j)}$ to 0 if this dimension is extra, which is equivalent to reducing m into $m-1$. In such a way, as long as the scale parameters in $\mathbf{k}$ are initially set to be larger than their appropriate values, $\mathbf{k}$ will be automatically driven to their appropriate values $\mathbf{k}^*$ during making parameter learning by $\max_\theta H(\theta, \mathbf{k})$ at fixed $\mathbf{k}$. It is easy to observe a significant reduction on the expensive computing cost of the two stage implementation in Fig.41, due to a reduction from repeating parameter learning for a large number of times into only implementing parameter learning one time. This is very appealing to those real applications that need fast implementation.


- Enumerate k for a set of candidate values, fixed at each candidate, make parameter learning

$\max_{\theta} H(\theta,k),\ \theta=\{\theta_1,\theta_2\}, \theta_2$ under certain constraint

Alternatively, we can also make

$\min_{\theta} KL(p\|q),\ KL(p\|q)=\int_{x,y} p(y|x)p(x)\ln\frac{p(y|x)p(x)}{q(x|y)q(y)}dxdy$

Select the best one k^* by

$k^*=\arg\min_k J(k),\ J(k)=-H(\theta^*,k)$

Fig. 44. Two step implementation

In the literature of a small sample size based statistical learning, it is well known that theoretical analysis is very difficult to conduct a comparative study on performances of different learning theories. Instead, this comparative study can be made via experiments. To facilitate this comparison, BYY harmony learning can also be made in a two stage implementation.

At the first stage, parameter learning can be made by $\theta^* = arg\max_\theta H(\theta, \mathbf{k})$ at a set of values of $\mathbf{k}$ with partial parameters in θ fixed, e.g., $\alpha_\ell = 1/k$ and the variance of $y^{(j)}$ is fixed at 1. Alternatively, parameter learning can be made by $\min_\theta KL(p\|q)$, which leads to $p(y|x)$ given by eq.(11) and further becomes equivalent to ML learning when $p(x)$ is given by the empirical density eq.(1).

At the second stage, model selection is made according to the following BYY criterion

$$\min_{\mathbf{k}} J(\mathbf{k}), \quad J(\mathbf{k}) = -H(\theta^*, \mathbf{k}). \tag{19}$$

Such a comparative study has been already made via some experiments [45], [46] on a Gaussian mixture structure in Fig.17 and on a Gaussian factor analysis structure in Fig.16(a). Experiments are made not only on simulated data sets of different sample sizes, noise variances, data space dimensions, and subspace dimensions, but also on two real data sets from air pollution problem and sport track records, respectively. Experiments have shown that BIC outperforms AIC, CAIC, and CV while the BYY criteria are either comparable with or better than BIC.

D. BYY harmony learning: (II) relations to other approaches

Denoting $p(u) = p(y|x)p(x), q(u) = q(x|y)q(y)$, we can rewrite eq.(18) into

$$H(p\|q) = \int p(u)\ln q(u)\mu(du) - \ln z_q. \tag{20}$$

When $\ln z_q = 1$, the above appearance is the 2nd part of Kullback information divergence $KL(p\|q) = \int p(u)\ln p(u)\mu(du) - \int p(u)\ln q(u)\mu(du)$, which has been studied in literatures for several decades. It looks not new and thus one may ask why the so called *Least complexity nature* of eq.(18) has not been discovered before.

The reason lays that generally considering $p(u), q(u)$ as a whole is very different from considering them in a Bayesian Ying Yang system. Considering $p(u), q(u)$ generally as a whole, we encounter two possibilities. One is $p(u) = \frac{1}{N}\sum_{t=1}^{N}\delta(u-u_t)$ given by an empirical density. In this case, $\int p(u)\ln q(u)\mu(du) = \frac{1}{N}\sum_{t=1}^{N}\ln q(u_t)$ is exactly the likelihood function that has been widely studied already. The other possibility is that $p(u)$ is free to be decided by $\max_p H(p\|q)$, which leads to a simplest form

$$p(u) = \delta(u-c), \ c \ is \ an \ arbitrary \ constant. \tag{21}$$

It is obviously useless and thus has not been further studied in the literatures.

In a contrast, considering the Ying Yang pair $p(u) = p(y|x)p(x), q(u) = q(x|y)q(y)$ with $p(x)$ given by eq.(1) or eq.(2) while $p(y|x)$ is either free or in a structural constraint, maximizing $H(p\|q)$ pushes $p(y|x)$ into a form of least complexity while maximizing $H(p\|q)$ pushes $q(x|y)q(y)$ to fit samples via matching $p(x)$ and also become a least complexity form via matching $p(y|x)$ that is pushed into a least complexity form. The details are referred to [108], [109], [103]. This is why BYY harmony learning got a model selection ability.

The appearance of $\int p(u)\ln q(u)\mu(du)$ may also lead to Akaike's AIC criterion that appears similar too. What is the difference ? In fact, Akaike's AIC criterion is derived from $\int p^*(u)\ln q(u)\mu(du)$ with $p^*(u)$ being the true density from which a set of samples $\{u_t\}_{t=1}^{N}$ comes. Also, this AIC only provides a criterion for implementing model selection at the 2nd stage of implementation, made after the parameters in $q(u)$ has been estimated via a ML learning. Actually, it not only has no automatic model selection ability, but also can not be used for parameter learning.

While AIC, as well as those existing learning theories discussed in Sec.III, considers $p(u), q(u)$ generally as a whole, BYY harmony learning considers $p(u) = p(y|x)p(x), q(u) = q(x|y)q(y)$ in a Ying Yang architecture. When $p(x)$ is given by eq.(1), ignoring $\ln z_q$ and assuming $N \to \infty$ we have

$$H(p\|q) = \int p(y|x)p^*(x)\ln[q(x|y)q(y)]\mu(dx)\mu(dy), \tag{22}$$

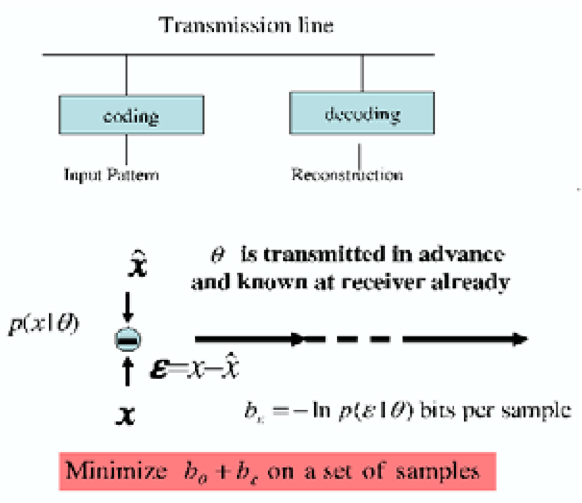
because $p(x)$ by eq.(1) tends to the true density $p^*(x)$ as $N \to \infty$. This $H(p\|q)$ can not be used for parameter learning since we do not know $p^*(x)$. However, we may use it to derive an improved model selection criteria to be used in a two stage implementation. One possibility is to use

$$p(y|x) = \delta(y - y(x)), \ y(x) = arg\max_y[q(x|y)q(y)]. \tag{23}$$

The other possibility uses $p(y|x)$ given by eq.(11). Moreover, we can also consider the expectation $E_X[H(p\|q)]$, where X is a set of samples on which parameters in $q(x|y)q(y)$ are obtained.

It should also be noted that BYY harmony based model selection applies to the problems of mining dependence structures that can be described in a Ying Yang architecture. There is problems it does not apply while AIC is applicable, e.g., determining the order p in eq.(12). In this sense, the applicable scope of AIC, as well as those existing learning theories, is wider than that of the BYY harmony learning.

The BYY harmony learning can also be understood from the perspective of optimal information transfer. As shown in Fig.45, a learning problem can be regarded as a problem that encodes information at the sending end, transfers the codes via transmission line, and then decodes the codes back to the original information. The principle is that encoding approach should be optimal for information transferring in a sense that the total number encoding bits is minimized, which has been widely studied in the past decade under the name of Minimum Description Length (MDL) [79], [80], [65], [66], [25], [41]. As previously discussed in Sec.III, the existing implementing approaches are actually equivalent to Bayesian inference criterion (BIC) [85], [55], [71].

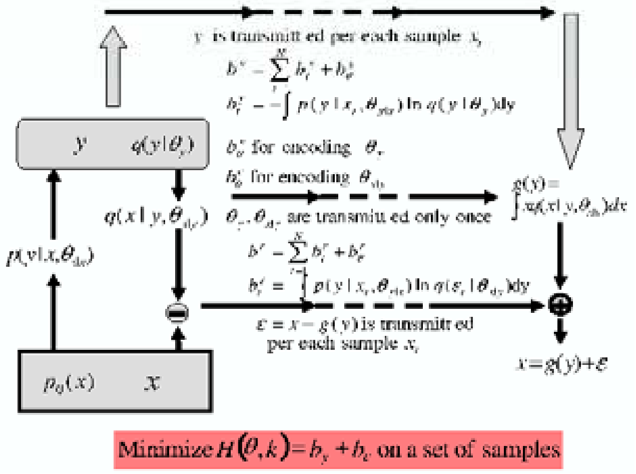


Transmission line
coding
decoding
Input Pattern
Reconstruction
$\hat{x}$
θ is transmitted in advance and known at receiver already
$p(x|\theta)$
$\varepsilon = x - \hat{x}$
x
$b_\varepsilon = -\ln p(\varepsilon|\theta)$ bits per sample
Minimize $b_\theta + b_\varepsilon$ on a set of samples

Fig. 45. Information transfer perspective and MDL principle

As shown in Fig.45, MDL makes encoding directly on samples of x. The total bits consist of two part. One is encoding each residual part b_ε of each sample that the model is unable to encode or fit, and this b_ε monotonically decreases as the scale parameter k increases. The other part b_θ is encoding the parameter set θ of the model. This b_θ contains the information of k and increases as the scale parameter k increases. The two parts trade off with an optimal k*. However, the part b_θ is difficult to estimate exactly and thus is roughly a much simplified term that relates only to k but not to θ, such that model selection is unable made during parameter learning.

As shown in Fig.46, the BYY harmony learning can also be viewed from an information transfer perspective. The key difference is that the part b_θ is replaced by $b_y = \sum_t b_t^y$ that encodes the inner representation y of each sample x. This b_y contains the information of k that is implied within each y. Moreover, b_y is a term that not only closely relates to θ but also is much easier to be estimated than b_θ. Further details are referred to [101], [109], where relations and key differences of BYY harmony learning have also been discussed together with many existing learning from approaches, including ML learning, Information geometry, Helmholtz machines, Variational approximation, Bit-back based MDL, etc.



y is transmitted per each sample x_t
$b^y = \sum_t^N b_t^y + b_\theta^y$
$b_t^y = -\int p(y|x_t,\theta_{y|x}) \ln q(y|\theta_y) dy$
y
$q(y|\theta_y)$
b_θ^y for encoding θ_y
b_θ^ε for encoding $\theta_{x|y}$
$q(x|y,\theta_{x|y})$
$\theta_y, \theta_{x|y}$ are transmitted only once
$g(y) = \int x q(x|y,\theta_{x|y}) dx$
$p(y|x,\theta_{y|x})$
$b^\varepsilon = \sum_t^N b_t^\varepsilon + b_\theta^\varepsilon$
$b_t^\varepsilon = -\int p(y|x_t,\theta_{y|x}) \ln q(\varepsilon_t|\theta_{x|y}) dy$
$\varepsilon = x - g(y)$ is transmitted per each sample x_t
$x = g(y) + \varepsilon$
$p_0(x)$
x
Minimize $H(\theta,k) = b_y + b_\varepsilon$ on a set of samples

Fig. 46. BYY harmony learning from an information transfer perspective

E. Model selection vs learning regularization

Regularization [91] and model selection are two different strategies for tackling the problem of a finite size of samples. Model selection prefers a model of least complexity for which a compact inner representation is aimed at, such that extra representation space can be released. In contrast, regularization is imposed on a model with a fixed scale of representation space with its complexity larger than needed such that inner representation can spread as uniformly as possible over the entire representation space with a distribution that is as simple as possible, which thus becomes equivalent to a model with a reduced complexity.

The harmony learning by eq.(18) attempts to compress the representation space via the least complexity that is demonstrated by eq.(23) which is actually a winner-take-all (WTA) competition. This type of parameter learning aims at a compact inner representation with an automatic model selection by discarding extra representation space during parameter learning. However, there is no free lunch. The WTA operation by eq.(23) locally per sample will make learning become sensitive to the initialization of parameters and the manner that samples are presented, resulting in that samples are over-aggregated in a small representation space. It usually leads to a local maximum solution for eq.(18).

With a soft competition by eq.(11) to replace the WTA competition by eq.(23), the ML learning, or $\min KL(p\|q)$, with a B-architecture and an empirical density by eq.(1), is regularized with a more spread inner representation that improves the local maximum problem. However, there is no free lunch. It makes the model selection ability considerably weaken, especially on a small size of samples. Thus, making model selection by eq.(19) is needed after parameter learning. Instead of the two phase style, regularization to the WTA by eq.(23) may also be imposed to the harmony learning by eq.(18), such that automatic model selection still occurs via either some external help or certain internal mechanism.

Externally, we can combine the learning by $\min KL(p\|q)$ with the learning by eq.(18), by which we get a spectrum of learning models. The details are referred to Sec. 23.4.2 in [107], [103]. Another spectrum, that also varies between model selection ability and regularization ability, can be obtained via internally replacing $\ln(r)$ by a family of convex functions for divergence measuring. Also, two different forms of the term $Z_q = -\ln z_q$ introduce two other types of regularization on learning under the name z-regularization. The details are referred to Sec.22.6.3 in [102].

Internally, regularization to the WTA by eq.(23) can be imposed during the harmony learning by eq.(18) via a constrained $p(y|x)$ in a BI-architecture. Instead of letting $p(y|x)$ free to be decided by eq.(23), we consider a BI-architecture with $p(y|x)$ designed in a structure that will not lead to the WTA by eq.(23). Specifically, a different structure of $p(y|x)$ will lead to a regularization with a different feature, which are shortly summarized under the name BI-regularization [104].

When y takes discrete values, instead of the winner-take-all, we can also consider "all the individuals of the winning team share the all", such that a local optimal problem can be alleviated.

$$\begin{aligned} & p(y|x_t) = a_0\delta(y-y_t) + \sum_{j=1}^{\kappa} a_j\delta(y-y_j), \\ & y_t = arg\max_y [q(x_t|y)q(y)], \\ & a_j > 0, j = 0, 1, \cdots, \kappa, \ \sum_{j=0}^{\kappa} a_j = 1. \end{aligned} \tag{24}$$

where $a_j > 0, j = 0, 1, \cdots, \kappa$ are pre-specified constants for the sharing percentages in the team, with a_0 being the largest. One possible team consists of the first κ largest winners of $\max_y[q(x_t|y)q(y)]$. The another possible team consists of the neighbors of the winner. The team may also consist of individuals with certain qualification similar to the winner.

When y takes real values, such a possible team may consists of an infinite members, e.g., we consider

$$p(y|x) = G(y|y(x), h_y^2 I), \ with \ y(x) \ by \ eq.(23), \tag{25}$$

for a given $h_y^2 > 0$ that can be determined in cooperation with a z_q-regularization. Togather with $p(x) = p_{h_x}(x)$ by eq.(2) and being put into eq.(18), we get

$$\begin{aligned} & H(\theta, m) = -\ln z_q(h_{x_2} h_y) + \\ & \tfrac{1}{N}\sum_{t=1}^{N} \int G(y|y(x), h_y^2 I) G(x|x_t, h^2 I) \ln[q(x|y)q(y)]dxdy \\ & = \tfrac{1}{N}\sum_{t=1}^{N} \ln[q(x_t|y(x_t))q(y_t)] + h_x^2 Tr[\frac{\partial^2 \ln Q(x)}{\partial x \partial x^T}]_{x=x_t} \\ & + h_y^2 Tr[\frac{\partial^2 \ln Q(y|x_t)}{\partial y \partial y^T}]_{y=y_t} - \ln z_q(h_x, h_y), \ y_t = y(x_t), \end{aligned}$$

$$Q(y|x) = q(x|y)q(y), \ Q(x) = q(x|y(x))q(y(x)). \tag{26}$$

In certain special cases, $y(x)$ by eq.(23) may have an analytic expression. One example is encountered when both $q(x|y)$ and $q(y)$ are Gaussian. In this case, it follows from eq.(23) that

$$y(x) = Wx + m. \tag{27}$$

In such a case, making the above learning by eq.(26) needs to consider the dependence from x to y as well as those parameters ψ in $y(x)$ in help of the chain rule. That is, when we make derivatives of a function in a form $F(x, y(x|\psi), \theta)$ with respect to x and ψ, we should consider

$$F'_x + \frac{\partial y^T(x|\psi)}{\partial x} F'_y, \quad \frac{\partial y^T(x|\psi)}{\partial \psi} F'_y, \tag{28}$$

where $f'_v = \frac{\partial f(u,v,w)}{\partial v}$ denotes the partial derivative of $f(u, v, w)$ with respect to the part u.

In general, $y(x)$ by eq.(23) is obtained by an optimization, it is difficult to have an analytic expression. In this case, we approximately consider $y_t = y(x_t)$ as a constant in $F(x_t, y_t, \theta)$ by ignoring its dependence to x and ψ, while certain regularization is still in action via h_x, h_y. Furthermore, we may consider to estimate $\frac{\partial y^T(x|\psi)}{\partial x}$ and $\frac{\partial y^T(x|\psi)}{\partial \psi}$, via discretely getting samples around x and ψ.

One special case of eq.(26) is considered by Eqn.(24) in [104]. Further ignoring the part of $q(y(x))$, another special case is considered by Eqn.(30) in [101]. Moreover, previous studies on data smoothing regularization are referred to Sec.2(B) in [110] and Sec.2.2.2 in [108], especially to a quite systematic summary in [106].

In addition to eq.(25), another possible team in the case of y taking real values can consist of merely several members, called *Competitive experts*. Considering to approximate the deterministic mapping function that has to be obtained by eq.(23) via optimization, we consider

$$\begin{aligned} & p(y|x) = \sum_{j=1}^{n} \beta_j(x)\delta(y - f_j(x, \theta_{y|x,j})), \\ & \sum_{j=1}^{n} \beta_j(x) = 1, \ \beta_j(x) = 0, \ or \ 1, \end{aligned} \tag{29}$$

from which eq.(23) is simplified into

$$\begin{aligned} & p(y|x) = \delta(y - y(x)), \ y(x) = f_{j^*(x)}(x|\theta_{y|x,j^*(x)}), \\ & j^*(x) = arg\max_j [q(x|y)q(y)]_{y=f_j(x|\theta_{y|x,j})}. \end{aligned} \tag{30}$$

Moreover, we can also consider eq.(25) with $y(x)$ given by eq.(30) and then put them into eq.(26).

Another example is using $p(y|x)$ in eq.(11), especially a Gaussian mixture when $q(x|y) = G(x|\mu_y, \Sigma_y)$, which was firstly proposed in [111] and has been further shown in [64] that this type of regularization actually performs a RPCL-like learning mechanism.

IV. Independent Subspace Learning and Extensions

In many real problems, an intrinsic dependence structure among a set of samples is usually of a much lower dimensional, especially for a small size of samples. Thus, projection of samples from the high dimensional observation space into a much lower dimensional space or manifold is a fundamental learning task. It has been widely studied in the fields of pattern recognition, machine learning, and image compression. A number of results have also been obtained by using BYY harmony learning on solving problems in this task.

A. *Fundamentals, linear projection, and least square error*

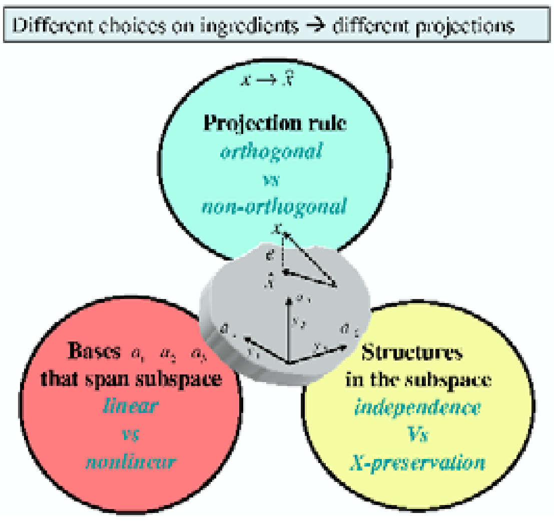


Fig. 47. Key Ingredients of subspace

1) Fundamentals: As shown in Fig.47, a projection task has three key ingredients. One is projection rule on how to map a sample in the high dimensional space to a low dimensional space. The other is certain structural constraints imposed on the subspace. Another is how the low dimensional subspace is represented. It can be spanned by a set of linear independent basis vectors $a_1, \cdots, a_m$. As shown in Fig.48, one sample x can have two types of projected representations. One is $\hat{x}$ in the original high dimensional space. The other is $y = [y^{(1)}, \cdots, y^{(m)}]^T$, the coordinates on the basis vectors $a_1, \cdots, a_m$. Two types of representations are related as follows:

$$\hat{x} = \sum_{j=1}^{m} a_j y^{(j)} = Ay, A = [a_1, \cdots, a_m], \tag{31}$$

where we have $det[A^TA] \neq 0$ since $a_1, \cdots, a_m$ are linear independent. With this link, a projection rule can be represented either within the original space in term a mapping $x \to \hat{x}$ or a mapping $x \to y$ also in help of the representation of the subspace.

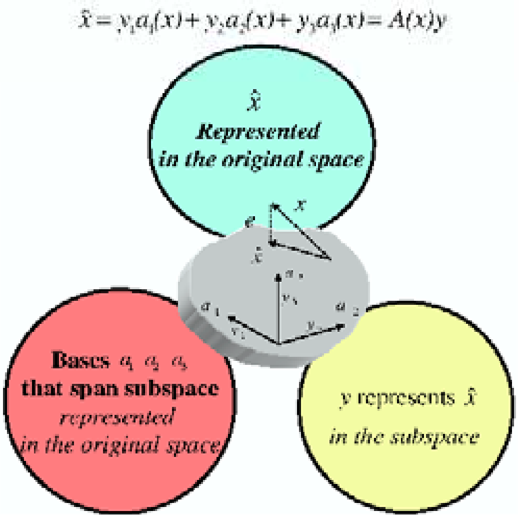


Fig. 48. Relation between two types of projected representations

These ingredients are closely related and different choices of their combinations will lead to different projections. The simplest projection rule is mapping a sample x into a point $\hat{x}$ on the subspace such that the distance $\|e\|^2$, $e = x - \hat{x}$ is shortest. This mapping $x \to \hat{x}$ is linear. For a linear representation by eq.(31), the mapping $x \to y$ is also linear. When the structure of subspace is flat, i.e., a subspace spanned by linear basis vectors $a_1, \cdots, a_m$, thus the shortest distance $\|e\|^2$ implies that the projection direction $x \to \hat{x}$ or equivalently the vector e is orthogonal to the subspace.

When the subspace is described by non-linear basis vectors $a_1(x), \cdots, a_m(x)$, the structure of subspace is curved. The shortest distance $\|e\|^2$ projection rule results in that $x \to \hat{x}$ and $x \to y$ may become nonlinear. Even for a subspace spanned by linear basis vectors $a_1, \cdots, a_m$, the shortest distance $\|e\|^2$ rule also leads to that $x \to \hat{x}$ and $x \to y$ may become nonlinear, when we consider either or both of the structure underlying samples in the original space and the structure of our interested distribution of y.

The structures that we are interested can be considered either from a perspective of seeking a most effective representation or a perspective of keeping certain natures of samples in the original high dimensional space, which leads to two streams of studies in the existing literatures.

One is featured by linear bases $a_1, \cdots, a_m$ plus requiring a least redundance in the representation y, i.e., components $[y^{(1)}, \cdots, y^{(m)}]^T$ are mutually independent. We refer studies of this stream by the name of independence subspaces. This stream can be traced back to the 30's-50's of the past century on principal component analysis (PCA) [42] and factor analysis in the literature of statistics

[6], [69], the 60's-80's on various subspace analysis in the literature of signal processing, and the past two decades up to now on adaptive PCA, independent component analysis (ICA)[68], [7], [128], [122], [124], [125], [126], [118], and independent factor analysis in the literature of neural networks and machine learning [68], [7], [114], [107].

The other stream is featured by a nonlinear projection rule plus requiring preservation of certain nature among samples in the original high dimensional space. The nature can be distances or topology between samples. This stream can also traced back to many previous efforts in the literature of statistics under the name *multidimensional scaling* [27], in the literature of neural networks under the name of *topological map* [57], [58], [14], and a recent renaissance [82], [90], [11].

Here we only focus on the stream of independence subspace.

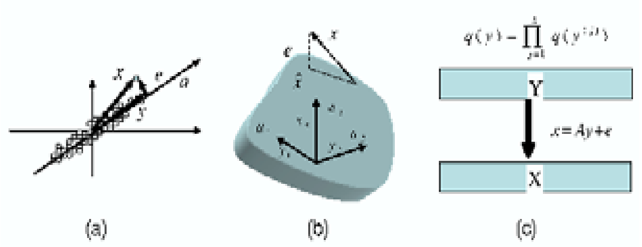


Fig. 49. Least square error projection

2) Linear projection $x \to y$ (I): Orthogonal $x \to \hat{x}$: We start at consider the simplest case of linear projection as shown in Fig.49. The subspace is spanned by only one basis vector a. We seek a linear projection $x \to \hat{x} = ya$, $y = a^T x$ on this subspace such that the residual or error $e = x - \hat{x}$ is minimized in the shortest distance sense, i.e., $\min_a \sum_t \|e_t\|^2$, which leads to that

$$Sa = \lambda_{max} a, \tag{32}$$

where S is the sample covariance matrix and λ_{max} is the largest eigenvalue of S, with the following nice properties:

$$\begin{array}{ll} (a) & \|a\| = 1 \text{ and } a \text{ is unique when } \lambda_{max} \text{ is unique.} \\ (b) & e \text{ and } y \text{ are not correlated, i.e.,} \\ & \sum_t e_t^T y_t = 0, \ or \ E[e^T y] = 0. \\ (c) & e \text{ is a Gaussian noise.} \end{array} \tag{33}$$

That is, it leads *principal component analysis* (PCA) [42] with $y = a^T x$ becoming the principal component,

In general, we consider a linear projection in the subspace by eq.(31) in the shortest distance sense:

$$\min_A \sum_t \|e_t\|^2, \ e = x - \hat{x}, \ y = A^T x, \tag{34}$$

which leads to that $a_1, \cdots, a_m$ span the same subspace spanned by the first m eigenvectors of S that correspond to the first m largest eigenvalues. That is, $a_1, \cdots, a_m$ span the principal subspace. The properties (b) & (c) in eq.(33) hold, while (a) becomes

$$\begin{array}{ll} (a.1) & \text{The subspace is unique when the } m\text{-th largest} \\ & \text{eigenvalue of } S > \text{the } m+1\text{-th largest;} \\ (a.2) & AA^T = I \text{ but } A \text{ is not unique, in a sense that} \\ & A = \phi\psi, \text{ where the column vectors of } \phi \text{ are} \\ & \text{the first } m \text{ eigenvectors of } S, \text{ but } \psi \text{ is an} \\ & \text{arbitrary } m \times m \text{ orthogonal matrix.} \end{array} \tag{35}$$

Among the infinite many solutions of A due to an arbitrary ψ, we may choose the one $\psi = I$, i.e.,

selecting m eigenvectors of S, corresponding to its first m largest eigenvalues, as the column vectors of A. (36)

In this case, we have that

$$\begin{array}{ll} (d) & \text{the components of } y = A^T x \text{ become} \\ & \text{uncorrelated, and the variance of each } y^{(j)} \text{ is} \\ & \text{one of the } m\text{-th largest eigenvalue of } S. \end{array} \tag{37}$$

In addition to the principal subspace, we also know the first m principal components. The task is also referred to m-PCA (or still PCA for simplicity).

As shown in Fig.49(a), minimizing $\|e\|^2$ is equivalent to maximizing $\|y\|^2$, when e is perpendicular to y. This happens when $\|a\| = 1$. That is, $\max_{a, \ s.t. \ \|a\|=1} E\|y\|^2$ also leads to PCA, which is actually where the PCA first came from. Similarly, from Fig.49(b) we also have such an equivalence when $A^T A = I$, that is, eq.(34) is equivalent to

$$\max_{A, \ s.t. \ A^T A = I} \sum_t \|y_t\|^2, \ y = A^T x. \tag{38}$$

3) Advances on Adaptive PCA and MCA : In 1982, Oja has mathematically proved that making a constrained Hebbian learning adaptively (i.e., per sample) will make $y = a^T x$ extract the first principal component [72]. This work, which is now referred as Oja rule, set up a connection between PCA and neural networks and trigged a lots of studies on implementing PCA related tasks via adaptive learning in the literature of neural networks for two decades. In 1989 [73], Oja further extended this Oja rule by using $y = Wx$ to extract the principal subspace. However, there is no proof on its global convergence though its local stability can be ensured via a mathematical analysis similar to that in [72]. By that time, it was regarded as being difficult to find a cost function from which the subspace rule can be derived [9]. Moreover, the Oja subspace rule can not realize eq.(37), i.e., performing m-PCA. Actually, by that time there was no adaptive rule available in the literature to perform m-PCA.

Solutions on these problems as well as further extensions have also been obtained since 1991 by the present author and colleagues, which are summarized as follows:

- In 1991, the present author derived a gradient descent iterative algorithm for solving the problem eq.(34) adaptively [144], [139], that is

$$\begin{aligned} &A^{new} = A^{old} + \eta_t[\delta x_t y_t^T + \delta x_t y_t^T], \\ &\hat{x}_t = A y_t, \ \hat{y}_t = A^T A y_t, \\ &\delta x_t = x_t - \hat{x}_t, \ \delta y_t = y_t - \hat{y}_t, \end{aligned} \tag{39}$$

  which has been shown be able to get the same principal subspace via setting up its link to the evolution direction of Oja subspace rule. Moreover, in help of this link, the global convergence of Oja subspace learning was, firstly in the literature, proved in [144], [139]. It was also pointed out in [139] that the Oja subspace learning is actually an adaptive version of the gradient ascent flow of eq.(38).
- Also in [144], [139], the weighting technique in the Brocket flow [19], that implements the orthogonal analysis on the Stiefel manifold $O(d,d)$, has been modified to the Stiefel manifold $O(d,m), m < d$ and thus be turned into an adaptive version such that both the Oja subspace rule and the above eq.(39) are improved into

$$\begin{aligned} &\hat{x}_t = W^T y_t, \ \hat{y}_t = W^T W y_t, \\ &W^{new} = W^{old} + \eta_t(z_t x_t^T - y_t \hat{x}_t^T), \\ &z_t = diag[d_1, \cdots, d_m] y_t \ with \ d_1 > \cdots > d_m, \\ &or \ W^{new} = W^{old} + \\ &\eta_t[(z_t x_t^T - y_t \hat{x}_t^T) + (z_t - \hat{y}_t) x_t^T], \end{aligned} \tag{40}$$

  which is the first result that can adaptively realize eq.(37, i.e., performing m-PCA [139].
- PCA is sensitive to outliers that will make the resulted a quite considerably deviated for the correct solution, as shown in Fig.50(a). Adaptive learning algorithms have been developed in [134] for implementing robust PCA and robust principal subspace in resistance of abnormal disturbances, as shown in Fig.50(b).
- Performing PCA learning in the cases that some elements of x are missing for certain samples [132]. For each of such sample $x = [x_i, x_o]$, the missing part x_i is recovered via the regression $\hat{x}_i = E(x_i|x_o)$, which can be computed from the covariance matrix Σ.

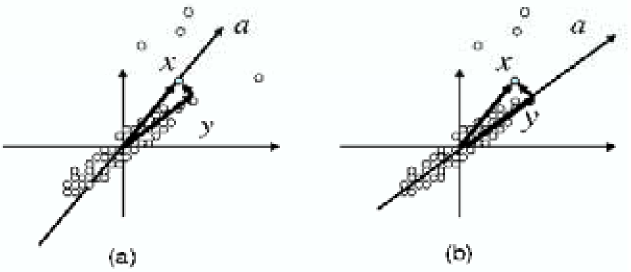


Fig. 50. Robust PCA

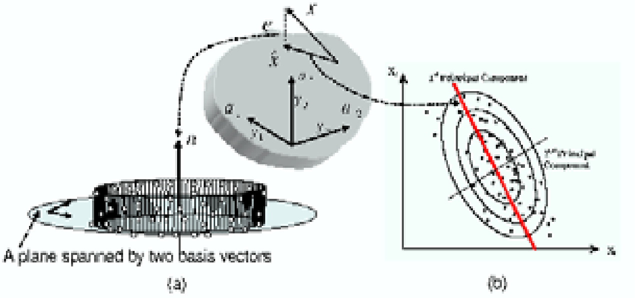


Fig. 51. PCA vs MCA

4) PCA vs Minor Component Analysis (MCA): As shown in Fig.50, when e is kept to be perpendicular to y or equivalently when $A^T A = I$ is kept, it is also meaningful to considering a dual problem, i.e., maximizing $\|e\|^2$ or minimizing $\|y\|^2$, which leads to a subspace that is the orthogonal complementary subspace, with a switching from seeking the part of largest eigenvalues to the part of smallest eigenvalues. Thus, the term 'principal' is switched to 'minor', that is, we have *Minor Component Analysis (MCA)* [144], [142].

Since 1991 the present author and colleagues have made the following contributions:

- Using its complementary subspace will be a more compact representation when the subspace dimension for describing a set of samples is larger than $0.5d$, e.g., as illustrated in Fig.50, when $d - m = 1$, finding $a^T x$ by MCA is equivalent to find the norm direction of a hyperplane that best fits a set of samples with the average of the distances of all the samples to the hyperplane being minimized. For this reason, we proposed, firstly in the literature, the so called dual subspace pattern recognition approach that uses both PCA and MCA to get subspaces for pattern classes [144].
- Though finding minor components has also been studied before 1992, e.g., in the Pisarenko method of spectral estimation [53], the name MCA was introduced to the field of neural networks firstly in [142], where an adaptive MCA algorithm is proposed and applied to the total least square fitting of a d-1 dimensional hyperplane.
- Also, nonlinearity has been introduced into the Hebbian learning not only via controlling the learning step size in an adaptive MCA but also via considering high order Hebbian learning in fitting a d-1 dimensional surface via transforming each nonlinear term $\prod_i x_i^{z_i}$, with z_i being an indicator taking either 0 or 1, into a

new variable u_j [142]. A complementary idea was also suggested in [112], [107] for using PCA to fitting a 1-dimensional nonlinear curve. Generally, a subspace obtained via PCA is used for fitting a nonlinear surface of less than $0.5d$ dimension, while the subspace obtained via MCA is used for fitting a nonlinear surface of larger than $0.5d$ [142].

- Conceptually, the Hebbian xy^T is related to PCA and the anti-Hebbian $-xy^T$ is related to MCA. This think tempted a quite number of studies either on investigating what will happen by simply changing the sign of learning for those existing PCA learning rules or on searching a unified updating rule that can implement PCA and MCA by simply changing the sign of learning [23]. Unfortunately, these efforts turned out few positive results. Actually, recalling our above discussion that switching from PCA to MCA is meaningful only when $A^TA = I$ is implicitly or explicitly satisfied, thus it is not well motivated to simply study changing the sign of learning, as been pointed out in [107]. With $A^TA = I$ in consideration, a general formulation, that enables to implement PCA and MCA by simply changing a sign, can be easily obtained [107].

Moreover, methods for making PCA with good performance under abnormal disturbance or missing data can be directly adopted for MCA. Also, the dimension m for PCA can be directly used to get the dimension $d - m$ for its complementary subspace.

B. Linear projection $x \to y$ (II): Nonorthogonal $x \to \hat{x}$

It follows from eq.(31) and eq.(34) that we can rewrite $x = \hat{x} + e = Ay + e$. Further considering the case $Ex \neq \mu$, we are lead to a same format of eq.(6). Precisely, we are lead to a special case of the linear latent structure as shown in Fig.(16)(a). Specifically, the coordinate vectors in a subspace act as a linear generative path, the representation y of the projection $\hat{x}$ on these coordinates act as inner factors stemming from which $\hat{x}$ is generated. Also, the projection residual or error e is regarded as observation noises added to $\hat{x}$ and thus we finally observe x.

However, y obtained via minimizing $\sum_t \|e_t\|^2$ does not satisfy eq.(4) automatically. It follows from eq.(35) that asking $y = A^Tx$ to satisfy eq.(4) is an additional requirement that needs an extra measure to ensure. When samples of x come from a Gaussian distribution, such a measure exists for the problem in Fig.49, e.g., by eq.(36). We call such a subspace by the name of *principal subspace*. This motivates us to further consider a subspace with its y satisfying eq.(4) and call it *independence subspace*. In the special case that $y = A^Tx$ satisfies eq.(37) or equivalently eq.(4) in a second order sense, we call it *de-correlated subspace*.

It has been further shown that minimizing $\sum_t \|e_t\|^2$, $e = x - AA^Tx$ in Fig.49 plus eq.(36) is actually lead to a solution A that is equivalent to the solution A of the ML learning on $q(x)$ by eq.(5) via the linear latent structure in Fig.16 and eq.(7) at the special case of $G(x|ADy, \sigma_e^2 I)$ with $A^TA = I, D = diag[d_1, \cdots, d_m]$.

The above connections motivate that not only those linear latent structures in Fig.(16) can be understood from a subspace perspective, but also *independence subspaces* can be further investigated via these linear latent structures from the perspective of BYY system and harmony learning.

1) Factor Analysis (FA): It has been widely studied and used in the literature of statistics and many other fields since 1956 [6], [69], and can be regarded as an extension of the previous principal subspace, by considering e coming from a Gaussian $G(e|0, \Sigma_e)$ in a general case. In this case, the project $x \to \hat{x}$ is still linear but its direction is no longer orthogonal to the subspace. As a result, the properties in eq.(33) and eq.(35) do not satisfy. To make the problem meaningful and solvable, we have to additionally impose not only the satisfaction of the property (b) in eq.(33), but also that y comes from $G(y|0, I)$. As a result, we consider the following model

$$q(x|y) = G(x|Ay, \Sigma_e), \; q(y) = G(y|0, I), \qquad (41)$$

the unknowns can be learned via a ML learning on $q(x)$ by eq.(5), namely $G(x|\mu, AA^T + \Sigma_e)$, which can be implemented by the EM algorithm [83].

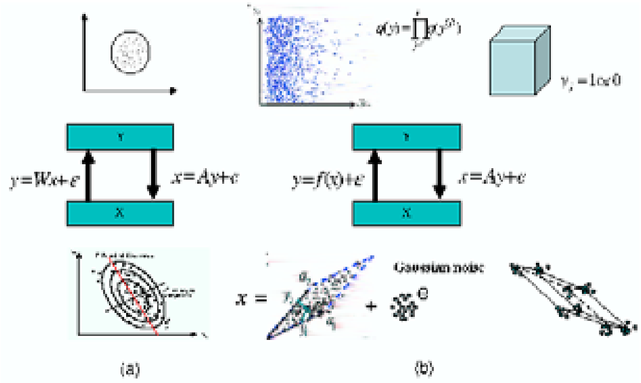


Fig. 52. Factor analysis in a bi-directional view

Further insights can be obtained from the perspective of BYY system in Fig.32. As previously stated already, the ML learning on $q(x)$ by eq.(5) is equivalent to the best matching learning $\min KL(p\|q)$ between the Ying-Yang pair by eq.(16), as shown in Fig.41, with $p(y|x)$ automatically determined as given by eq.(11). In such a sense, factor analysis in a backward architecture as shown in Fig16(a) is equivalent to making factor analysis via a bi-directional architecture as shown in Fig.52(a). From this perspective, one adaptive EM algorithm is developed in

Sec.4.2.5 in [114] and also a variant algorithm is developed in eqn.(32) of [116].

Moreover, we also observe the following interesting natures:

$$\begin{array}{ll} (e) & \text{the forward path is linear } y = W(x-\mu)+\varepsilon\ , \\ & \ \text{with } x \text{ and } \varepsilon \text{ uncorrelated,} \\ (f) & p(\varepsilon) = G(\varepsilon|0,\Sigma_\varepsilon),\ W = A^T(AA^T+\Sigma_e)^{-1}, \\ & \Sigma_\varepsilon = I - A^T(AA^T+\Sigma_e)^{-1}A, \\ (g) & E(yy^T) = WSW^T + \Sigma_\varepsilon. \end{array} \qquad (42)$$

where S is the sample co-variance matrix, as the sample size $N \to 0$, both S and $AA^T + \Sigma_e$ tend to the true co-variance matrix of x, and thus $E(yy^T) = WSW^T + \Sigma_\varepsilon \to I$. In order word, the forward path $y = W(x-\mu)+\varepsilon$ maps samples of x into y that satisfies the pre-specified assumption $q(y) = G(y|0,I)$ as $N \to 0$. However, when N is a finite or small size, not only the satisfaction of $G(y|0,I)$ does not hold, but also the satisfaction of eq.(4) also no longer holds.

It is interesting to further observe the special case of eq.(7) at $G(x|ADy, \sigma_e^2 I)$ with $A^TA = I, D = diag[d_1, \cdots, d_m]$. In this case, the ML learning on $G(x|\mu, AD^2A^T + \sigma_e^2 I)$ with $A^TA = I, D = diag[d_1, \cdots, d_m]$ leads to that $d_1^2, \cdots, d_m^2$ and the columns of A are the first m eigenvalues and the corresponding eigenvectors of S. Not only the solution A is, as discussed previously, equivalent to that obtained by minimizing $\sum_t \|e_t\|^2,\ e = x - AA^Tx$ plus eq.(36), but also $y = W(x-\mu)+\varepsilon$ maps samples of x, similar to the map $y = A^Tx$, into y that satisfies eq.(4) without $N \to 0$, though this may not satisfy $q(y) = G(y|0,I)$. This point can be observed from eq.(42) that not only we have $(AD^2A^T + \sigma_e^2 I)^{-1} = \sigma_e^{-2}[I - A(\sigma_e^2 D^{-2} + I)^{-1}A^T]$ and $\Sigma_\varepsilon = I - A^T(AA^T + \Sigma_e)^{-1}A = I - \Pi$, where $\Pi = \sigma_e^{-2}[I - (\sigma_e^2 D^{-2} + I)^{-1}]$ is diagonal, but also we have $W = A^T(AA^T + \Sigma_e)^{-1} = \Pi A^T$ and $WSW^T = \Pi A^T SA\Pi = \Pi\Lambda\Pi$, where $\Lambda = A^TSA$ is a diagonal matrix consisting of the first m eigenvalues of S.

When $\Sigma \neq \sigma^2 I$, however, the FA model suffers a serious indeterminacy problem due to two types of indeterminacy:

- *rotation indeterminacy*, i.e., $Eyy^T = Ey'y'^T = I$ and $A\phi^T\phi A^T = AA^T$ for any $y' = \phi y, \phi^T\phi = I$.
- *additive indeterminacy*, i.e., $AA^T + \Sigma = A'A'^T + \Sigma'$ for any C such that $AA^T + C = A'A'^T$ and $\Sigma' = \Sigma - C$ remains to be nonnegative definite.

Due to the indeterminacy, the ML learning or equivalently the best matching learning $\min KL(p\|q)$ has an infinite number of solutions. Considering the singular value decomposition

$$\begin{array}{l} A = \phi D\psi^T,\ \psi^T\psi = I,\ \phi^T\phi = I, \\ D = diag[d_1, \cdots, d_m], \end{array} \qquad (43)$$

we can observe that the FA model by eq.(41) is actually equivalent to the following model:

$$q(x|y) = G(x-\mu|\phi Dy, \Sigma_e),\ q(y) = G(y|0,I), \qquad (44)$$

in the sense of the ML learning on $q(x)$ by eq.(5).

At a given subspace dimension m, the behavior of the BYY harmony learning is same as the ML learning on $q(x)$ by eq.(5) asymptotically as $N \to \infty$. However, for a finite N and a unknown m, a salient feature of BYY harmony learning is able to determine an appropriate m.

For a two stage implementation of learning, BYY criteria are obtained from eq.(19) for selecting m, as summarized in Tab.2 of [107]. Recently, an improved BYY criterion is also derived in [44] by considering the role of $y(x) = Wx + \mu$ in $h_x^2 Tr[\frac{\partial^2 \ln Q(x)}{\partial x \partial x^T}]_{x=x_t}$ in Eqn.(24) of [104], which is a special case of eq.(26).

Moreover, adaptive algorithms have been developed for implementing learning, with an appropriate m determined automatically during learning. One is considering eq.(44) by iterating

$$\begin{array}{ll} (a) & y_t = [I + D\phi^T\Sigma_e^{-1}\phi D]^{-1}D\phi^T\Sigma_e^{-1}(x_t - \mu), \\ (b) & e_t = x_t - \phi Dy_t,\ g_\phi = \Sigma_e^{old\ -1}e_t y_t^T D^{old}, \\ & \phi^{new} = \phi^{old} + \eta(g_\phi - \phi^{old} g_\phi^T \phi^{oldT}), \\ & D^{new} = D^{old} + \eta diag[y_t e_t^T \Sigma_e^{old\ -1}\phi^{old}], \\ & \mu^{new} = \mu^{old} + \eta e_t, \\ & \Sigma_e^{new} = (1-\eta)\Sigma_e^{old} + \eta e_t e_t^T. \end{array} \qquad (45)$$

This learning determines m via minimizing $-\ln G(y|0,I) = c + 0.5\|y\|^2$ and thus indirectly pushing $y^{(j)\ 2}$ (and thus d_j) of an extra dimension to 0. It also provides an alternative algorithm for implementing m-PCA when $\Sigma_e = \sigma_e^2 I$.

It can also be further observed that eq.(44) can be turned into the following equivalent model:

$$q(x|y) = G(x-\mu|\phi y, \Sigma_e), q(y) = G(y|0, D^2), \qquad (46)$$

in the sense of the ML learning. In this case, maximizing $\ln G(y|0,D^2)$ or equivalently minimizing $\ln|D|^2 + 0.5y^TD^{-2}y$ tends to push $y^{(j)\ 2}$ with a small d_j toward zero and thus d_j becomes even smaller, which speed up to push d_j of an extra dimension to 0. The implementation can be made by iterating

$$\begin{array}{ll} (a) & y_t = [I + \phi^T\Sigma_e^{-1}\phi]^{-1}D^{-1}\phi^T\Sigma_e^{-1}(x_t - \mu), \\ (b) & e_t = x_t - \phi y_t,\ g_\phi = \Sigma_e^{old\ -1}e_t y_t^T, \\ & update\ \phi, \Sigma_e, \mu\ as\ in\ eq.(45), \\ & D^{2\ new} = D^{2\ old} + \eta diag[y_t y_t^T]. \end{array} \qquad (47)$$

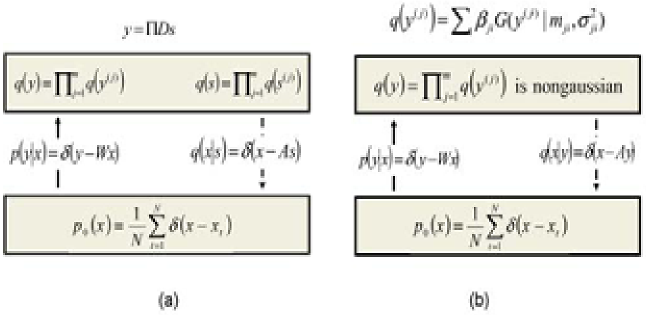
In the case of a small size of samples, taking the point of eq.(28) in consideration, we can combine (a)&(b) in eq.(47) together and get an algorithm that update A, D, Σ_e to increase (e.g., along the gradient ascend direction)

$$\begin{array}{l} [\ln G(x_t - \mu|\phi y, \Sigma_e) + \ln G(y_t|0, D^2)], \\ s.t.\ y_t = [I + \phi^T\Sigma_e^{-1}\phi]^{-1}D^{-1}\phi^T\Sigma_e^{-1}(x_t - \mu). \end{array} \qquad (48)$$

2) Independent Component Analysis (ICA): The rotation indeterminacy can be removed on samples of x that contains more structures. One possibility is that samples are generated from nonGaussian factors of y that satisfies eq.(4). Due to implementing difficulty in the case $e \neq 0$, most studies in the literature for past two decades have been made in a much simplified case with $e = 0$, i.e., $x = Ay$. In this case, we know that there exists at least one A such that it follows from eq.(31) that $\hat{x} = x$. An identity mapping can reach this. That is, we want $x = Ay = AWx$ with $AW = I$. That is, the problem of minimizing projection error becomes finding the inverse of a unknown A. Given a y satisfying eq.(4) and with at most one of components is gaussian, any permutation Π and constant scaling D of components will give a $y' = \Pi D y$ that still satisfies eq.(4) [92]. In other words, $y = Wx$ that satisfies eq.(4) leads to $AW = \Pi D$, i.e., the inverse of A is found up to a permutation and constant scaling. Thus, the problem becomes getting a linear projection $y = Wx$ as shown in Fig.14(b) such that the components of y become component-wise independent, which have been extensively studied in the past two decades under the name of independent component analysis (ICA) [50], [26].

Although ICA has been studied from different perspectives, such as the minimum mutual information (MMI) [12], [5] and maximum likelihood (ML) [36], all the approaches are equivalent to maximizing the following cost

$$L(W) = \ln|W| + L_y(W),$$



$y=\Pi Ds$
$q(y)=\prod_{j=1} q(y^{(j)})$
$q(s)=\prod_{j=1} q(s^{(j)})$
$p(y|x)=\delta(y-Wx)$
$q(x|s)=\delta(x-As)$
$p_0(x)=\frac{1}{N}\sum_{t=1}^{N}\delta(x-x_t)$
(a)
$q(y^{(j)})=\sum \beta_{jt} G(y^{(j)}|m_{jt},\sigma_{jt}^2)$
$q(y)=\prod_{j=1}^{m} q(y^{(j)})$ is nongaussian
$p(y|x)=\delta(y-Wx)$
$q(x|y)=\delta(x-Ay)$
$p_0(x)=\frac{1}{N}\sum_{t=1}^{N}\delta(x-x_t)$
(b)

Fig. 53. Independent component analysis from a bi-directional view

As shown in Fig.53(a), the ICA problem can also be viewed from the perspective of BYY system in Fig.32. Not only we observed that the ML learning on $q(x)$ by eq.(5), the best matching learning $\min KL(p\|q)$ between the Ying-Yang pair by eq.(16), and BYY harmony learning are all equivalent; but also we obtained several new results as follows:

- Eq.(49) is further extended into [126], [115], [63], [78]:
$$L(W) = 0.5\ln|WW^T| + L_y(W),$$
$$or \quad L(W) = L_y(W)_{s.t.} \ W \text{ is a full rank}, \tag{50}$$
for the cases of $m \leq d$. Moreover, eq.(49) and eq.(50) have been further extended with $\ln(r)$ replaced by a generalized convex function $f(r)$ that has been shown experimentally to be more robust to outliers [126].
- As shown in Fig.53(b), each model pdf $q(y^{(j)})$ is suggested to be a flexibly adjustable density that is learned together with W, with the help of either a mixture of sigmoid functions that learns the cumulative distribution function (cdf) of each source [128], [124], [125], [126] or a mixture of parametric pdfs [122], [118], and a so-called learned parametric mixture based ICA (LPMICA) algorithm is derived, with successful results on sources that can be either subgaussian or supergaussian, as well as any combination of both types. The mixture model was also adopted in the ICA algorithm by [74], although it did not explicitly target at separating the mixed sub- and supergaussian sources.
- It has also been found that a crude estimate of each source pdf or cdf may already be enough for source separation. For instances, a simple sigmoid function such as $tanh(x)$ seems to work well on the supergaussian sources [12], and a mixture of only two or three gaussians may be enough already [118] for the mixed sub- and supergaussian sources. This leads to the so-called one-bit-matching conjecture [119], which states that all the sources can be separated as long as there is an one-to-one same sign- correspondence between the kurtosis signs of all source pdfs and the kurtosis signs of all model pdfs. In past years, this conjecture has also been implicitly supported by several other ICA studies [37], [35], [59], [96]. In [24], a mathematical analysis was given for the case involving only two subgaussian sources. In [4], stability of an ICA algorithm at the correct separation points was also studied via its relation to the nonlinearity $\phi(y^{(j)}) = d\ln q(y^{(j)})/dy^{(j)}$, but without touching the circumstance under which the sources can be separated.
- In [60], the one-bit-matching conjecture on multiple sources has been proved mathematically in a weak sense. When only sources' skewness and kurtosis are considered with $Es = 0$ and $Ess^T = I$, and the model pdf's skewness is designed as zero, it is further proved that the one-bit-matching conjecture is true when the global maximum of eq.(49) with respect to W is reached. However, this proof still can not support the successes of many existing iterative ICA algorithms that typically implement gradient based local search

and thus usually converge to one of local optimal solutions.

- Recently in [99], a new mathematical proof is obtained in a strong sense that the conjecture is also true when anyone of local optimal solutions is reached, in help of investigating a convex-concave programming on a polyhedral-set. Theorems have also been provided not only on partial separation of sources when there is a partial matching between the kurtosis signs, but also on an interesting duality with super-gaussian sources separated via maximization and sub-gaussian sources separated via minimization. Moreover, a corollary is a obtained to confirm the symmetric orthogonalization implementation of the kurtosis extreme approach for separating multiple sources in parallel, which works empirically but still in a lack of mathematical proof [48].

There are some studies on determining the number m of independent components in the literature. Strictly speaking, this issue is trivial for ICA when $e = 0$, since the dimension of y is the number of nonzero eigenvalues of the sample covariance matrix S. However, all the eigenvalues of S will be nonzero when there is a noise e, which is beyond the scope of ICA and will be discussed in the next subsection.

Also, it deserves to mention that there is a widespread misunderstanding in the current ICA literature that ICA is a generalized counterpart of PCA [26]. Precisely, it is not true since indeterminacy on scalings makes meaningless to make a selection according to variances of the components of y. A generalized counterpart of PCA should take $e \neq 0$ in consideration. One example is the previous FA, while other examples will be also discussed later.

As pointed in [107], ICA is actually an generalized counterpart of *De-correlating analysis (DCA)*, i.e., a linear mapping $y = Wx$ from x to y with its components being mutually independent in the 2nd orders or called de-correlated. One particular case is becoming $I = Eyy^T = WExx^TW^T$, as shown in Fig.14(a), which is referred as a whitening transform. Moreover, it follows from $I = WExx^TW^T$ that $I = \tilde{W}Exx^T\tilde{W}^T, \tilde{W} = \phi W$ for any $\tilde{W}$ with $\tilde{W}Exx^T\tilde{W}^T = I$. In other words, W is subject to an indeterminacy of an orthogonal matrix.

A generalized counterpart of whitening transform, with the above rotation indeterminacy removed, is a particular case of ICA as shown in Fig.14(c). If x is mapped into a uniform distribution, $f(x)$ is the cumulated distribution function (CDF) of x and we get an estimate on the density of x as follows [107].

$$q(x) = |W(x)W^T(x)|^{0.5}, \ W(x) = \frac{\partial f(x)}{\partial x^T}. \qquad (51)$$

In implementation, $f(x)$ can be a weighted sum of sigmoid functions. At the end of Sec.4 of [107], extension has also been made to the case that the dimension d of x is larger than the dimension m of y.

3) Temporal de-correlating analysis (T-DCA) and Temporal Factor Analysis (TFA) : A generalized counterpart of DCA, with the rotation indeterminacy removed, is called Temporal DCA (T-DCA) by considering temporal relation within observations in a sense that temporal observations of x_t is mapped still by a linear mapping into a temporal model of y_t with its components mutually de-correlated at every time. E.g., in the simplest case, we consider the linear mapping $y_t = Wx_t$ together with the following linear temporal model

$$y_t = By_{t-1} + \varepsilon_t, \qquad (52)$$

where ε_t is a white noise with $E\varepsilon_t = 0$ and $E\varepsilon_t\varepsilon_\tau^T = \delta(t-\tau)$, and B is a diagonal matrix with each diagonal element $|b_j| < 1$ to ensure the stability of the above 1st order AR model. It follows from eq.(52) that $\Lambda = B\Lambda B + I$ and thus $B^2 = I - \Lambda^{-1}$, as well as it follows from $Wx_t = BWx_{t-1} + \varepsilon_t$ that $WE[x_tx_{t-1}^T]W^T = BWE[x_{t-1}x_{t-1}]^TW^T = B\Lambda$ and thus $WE[x_tx_{t-1}^T]W^T$ is also diagonal. That is, W should make both $E[x_tx_{t-1}^T]$ and $E[x_{t-1}x_{t-1}^T]$ become diagonal simultaneously. Except the degenerated case that x_t, x_{t-1} are i.i.d. (in this case $B = 0$), there will be a unique and non-orthogonal W that can make this simultaneous diagonalization. That is, there will be no indeterminacy of a rotation matrix for a T-DCA. Moreover, learning on eq.(52) can be made adaptively per sample as follows:

$$\begin{aligned} &\varepsilon_t = Wx_t - By_{t-1}, \ W^{new} = W^{old} + \eta\varepsilon_t x_t^T, \\ &B^{new} = B^{old} + \eta diag[\varepsilon_t y_{t-1}^T]. \end{aligned} \qquad (53)$$

Also, the above 1st order AR on y_t can be extended to a higher order AR.

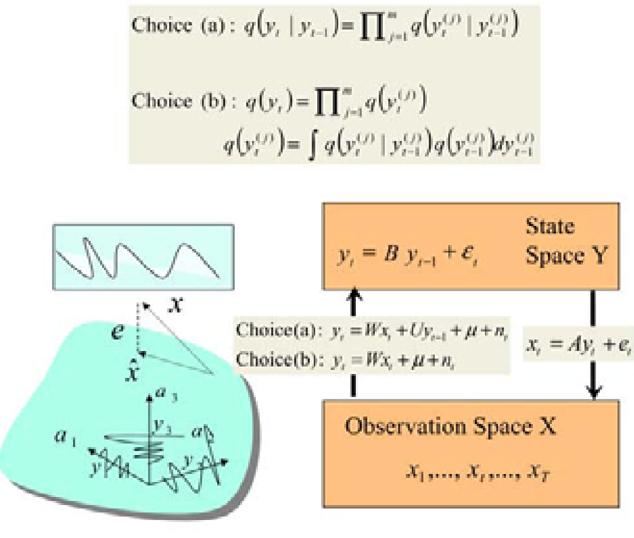

Choice (a) : $q(y_t \mid y_{t-1}) = \prod_{j=1}^m q(y_t^{(j)} \mid y_{t-1}^{(j)})$
Choice (b) : $q(y_t) = \prod_{j=1}^m q(y_t^{(j)})$
$q(y_t^{(j)}) = \int q(y_t^{(j)} \mid y_{t-1}^{(j)}) q(y_{t-1}^{(j)}) dy_{t-1}^{(j)}$
x
e
$\hat{x}$
a_3
a_1
a_2
y_1
y_2
y_3
$y_t = B\,y_{t-1} + \varepsilon_t$
State Space Y
Choice(a): $y_t = Wx_t + Uy_{t-1} + \mu + n_t$
Choice(b): $y_t = Wx_t + \mu + n_t$
$x_t = Ay_t + e_t$
Observation Space X
$x_1, \ldots, x_t, \ldots, x_T$

Fig. 54. Temporal Factor analysis (TFA)

One other generalized counterpart of PCA, that takes $e \neq 0$ in consideration, is a temporal extension of FA, called temporal FA (TFA). When there is certain temporal relation among samples of x, as shown in Fig.54, we can correspondingly use a temporal structure, e.g., eq.(52), to describe y. Again due to a scaling indeterminacy $Ay_t = A'y'_t$ with $y' = Dy$, we consider Gaussian $G(y_t|By_{t-1}, I)$, where $B = diag[b_1, \cdots, b_m]$ with each $|b_j| < 1$, that is, factor analysis is extended into the temporal model by eq.(15), which is proposed firstly in [113], [110] under the name of temporal factor analysis (TFA). Here, we rewrite it below:

$$x_t = Ay_t + \mu + e_t, \; y_t = By_{t-1} + \varepsilon_t. \qquad (54)$$

where y_t, e_t are mutually independent, y_{t-1}, ε_t are mutually independent, the components of y_t are uncorrelated. Moreover, A, B, the covariance matrix Σ_e of e_t are all unknown.

In this TFA model, a rotation $y'_t = \phi y_t$ will lead to $G(y'_t|\phi B \phi^T y'_{t-1}, I)$ with $\phi B \phi^T = B'$ no longer diagonal. Thus, the m channels are no longer independent. In other words, the rotation indeterminacy of FA has disappeared in TFA due to considering the linear and first order temporal relation $y_t = By_{t-1} + \varepsilon_t$. Moreover, samples of x are described by $G(x_t|\mu, A\Lambda_t A^T + \Sigma_e)$ with $E[y_t y_t^T] = \Lambda_t$. The additive indeterminacy between the two parts $A\Lambda_t A^T$ and Σ can be reduced or removed by the structure $\Lambda_t = B\Lambda_{t-1}B + I$.

As shown in Fig.54, the problem of this TFA model can also be regarded as setting up a temporal independence subspace. That is, projecting a time series x_t into a representation $\hat{x}_t$ in a subspace such that $\hat{x}_t$ is represented by several temporal components in this subspace. As shown in Fig.48, the linear representation $\hat{x}_t = Ay_t$ and the error $e_t = x_t - \hat{x}_t$ only depend on the current time t. As shown in Fig.54, temporal information is transferred via the past value y_{t-1} in either of the following two choices:

$$\begin{aligned} &(a) \quad q(y_t|y_{t-1}), \\ &(b) \quad q(y_t) = \int q(y_t|y_{t-1}) q(y_{t-1}) dy_{t-1}. \end{aligned} \qquad (55)$$

As shown in Fig.54, the situation is basically same as the previously discussed factor analysis for the choice (b). The only difference is that $E[y_t y_t^T] = \Lambda_t$ varies with t. For the choice (a), regarding $By_{t-1} = \nu_t$ as the mean, we can rewrite $G(y_t|By_{t-1}, I)$ into $G(y_t|\nu_t, I)$, the learning problem is also similar to FA, except that we need to additionally consider the learning on B.

Further details about TFA are referred to [113], [110], especially to [107], [100], where several results about implementing TFA learning can be found, including:

- Adaptive learning algorithms for both types by eq.(55), with the dimension m determined automatically;
- Criteria for selecting m in a two stage implementation of learning;
- Relation to the Kalman filtering and system identification in the state space by eq.(54);
- Applications in modelling financial market.

C. Beyond linear projection $x \rightarrow y$: NFA, BFA, and LMSER

Another generalized counterpart of PCA is considering that samples of x are not from Gaussian and but generated via $x = Ay + e, e \neq 0$ from y by eq.(4) with each $q(y^{(j)})$ being nonGaussian. A rotation $y'_t = \phi y_t$ with $\phi \neq I$ will make mutually independent components of y_t become coupled. In other words, the rotation indeterminacy disappears. This point can also be observed from that we can also get extra constraint equations by considering higher order statistics on both the sides of eq.(6), in addition to the constraint $S = AA^T + \Sigma_e$ only for the case of FA model. Moreover, the extra equations may also make the additive indeterminacy removed.

As shown in Fig.55, each nonGaussian density $q(y^{(j)})$ imposes certain structure on each coordinate of the subspace, which makes the subspace becomes curved. Thus, the projection $x \rightarrow \hat{x}$ and the projection $x \rightarrow y$ both become nonlinear. This makes the problem of jointly setting up this subspace and finding such a projection very difficult. The integral in eq.(5) becomes not analytically tractable, and thus $p(y|x)$ by eq.(11) is also not analytically computable. A straight forward way is an approximation method such as a Monte Carlo random sampling approach [114]. However, the computation is usually very involved. As a result, it is very difficult to make the ML learning, and thus this problem has been seldom studied in past decades.

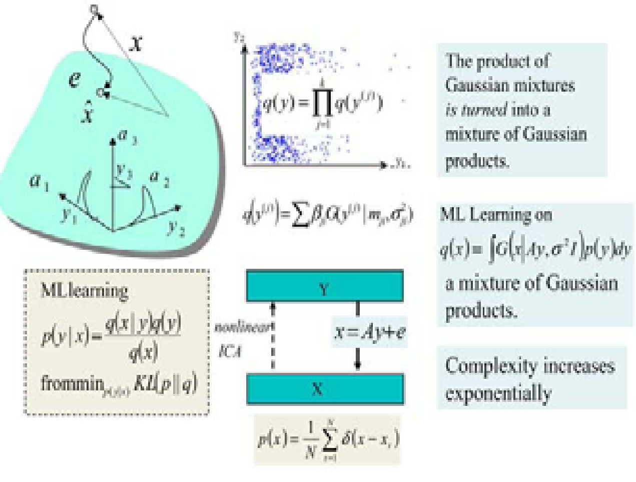

x
e
$\hat{x}$
a_3
a_1
y_3
a_2
y_1
y_2
y_2
y_1
$q(y) = \prod_{j=1}^{k} q(y^{(j)})$
The product of Gaussian mixtures *is turned* into a mixture of Gaussian products.
$q(y^{(j)}) = \sum \beta_{ji} G(y^{(j)}|m_{ji}, \sigma_{ji}^2)$
ML Learning on
$q(x) = \int G(x|Ay, \sigma^2 I) p(y) dy$
a mixture of Gaussian products.
ML learning
$p(y|x) = \frac{q(x|y)q(y)}{q(x)}$
from $\min_{p(y|x)} KL(p\|q)$
Y
nonlinear ICA
$x = Ay + e$
X
Complexity increases exponentially
$p(x) = \frac{1}{N}\sum_{t=1}^{N} \delta(x - x_t)$

Fig. 55. ML learning by an EM algorithm

As shown in Fig.55, an approach that can make ML learning on eq.(6) was firstly proposed in [68]. Echoing [122], they considered the independence product eq.(4)

with each $q(y^{(j)})$ modelled by a Gaussian mixture. A key difference is that they dealt with the product of Gaussian mixtures via introducing a set of random variable $z^{(j)}, j = 1, \cdots, m$ such that each $z^{(j)}$ stochastically takes a number among $\{1, \cdots, n_j\}$ and each number indicates a component in the j-th mixture. Thus, it follows from $q(y) = \sum_{\{z^{(j)}\}} q(y, z)$ that the product of m summations in eq.(4) is equivalently exchanged into a summation of $\prod_j n_j$ products. As a result, the integral on getting $q(x)$ becomes a summation of $\prod_j n_j$ analytically computable integrals on Gaussians, which results in that $q(x)$ becomes a mixture of $\prod_j n_j$ Gaussians. For this reason, they were able to implement a ML learning by an EM algorithm. A same result has been also published in [7] under the name of independent factor analysis. A serious problem of their work [68], [7] is that a summation of $\prod_j n_j$ terms has to be computed at each step of such an EM algorithm. The complexity increases exponentially with the number m of factors, i.e., $O(n^m)$ with $n = \max_j n_j$.

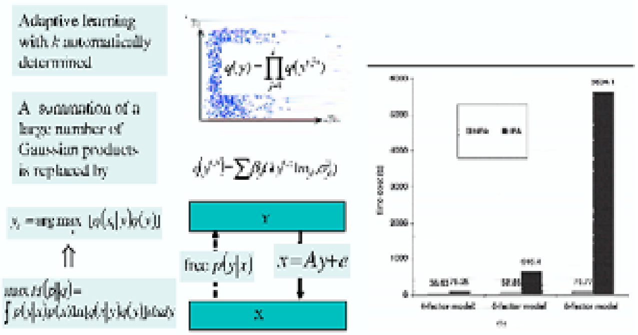

Adaptive learning with k automatically determined
A summation of a large number of Gaussian products is replaced by
$q(y) = \prod_{j=1} q(y^{(j)})$
$q(y^{(j)}) = \sum \beta_j G(y^{(j)}|m_j, \sigma_j^2)$
Y
X
max p(y|x)
x = Ay + e
max H(p||q) =
BNFA
BFA
3-factor model
5-factor model
6-factor model

Fig. 56. NonGaussian factor analysis (NFA)

In help of BYY harmony learning by eq.(18), the problem has been also studied under the name of nonGaussian factor analysis (NFA) [110], [107]. As shown in Fig.56, the difficulty of making the integral over y in the whole domain R^m have been turned into the problem of a nonlinear optimization solved by an iterative algorithm that searches within the domain R^m in a trace that is usually within a subspace of much lower dimension. It has been experimentally shown that its computing complexity increase with m linearly.

Moreover, not only criteria have been obtained for selecting m, but also an adaptive algorithm has been proposed under the name of uncorrelated NFA, such that the subspace dimension can be determined automatically during learning. In help of the SVD by eq.(43), the problem is turned into [107]:

$$q(x|y) = G(x|\phi y + \mu, \Sigma_e), \ \phi^T \phi = I,$$
$$q(y) = |D| q_y(\xi), \ \xi = \psi D^{-1} y, \ \psi^T \psi = I, \quad (56)$$

where $q_y(\xi)$ is a parametric model that satisfies eq.(4). It still falls in the paradigm of the conventional factor analysis [69] and thus we call it uncorrelated NFA by which ξ_t acts as the recovered independent factors though y_t is not. That is, $x_t \to \xi_t$ performs an independence mapping that takes in consideration of the noise e_t. The details are referred to [107]. In Sect. 4.2 of [101], this approach has been further extended to also automatically determining the number of Gaussians in every mixture $\sum_t \beta_{jt} G(\varepsilon^{(j)}|m_{jt}, \lambda_{jt}^2)$.

We can further combine the situations when both samples of x are not from Gaussian and there is temporal relation among samples of x, such that not only rotation indeterminacy and additive indeterminacy can be removed but also dependence structure among samples of x can be better described. One example of such a combination is considering eq.(54) with Gaussian $G(\varepsilon_t|0, I)$ replaced by a nonGaussian density for describing ε_t in a way similar to the above case for y. The details are referred to [100].

When each $y^{(j)}$ is a binary number that comes from a Bernoulli distribution by eq.(8), eq.(6) has been studied under the name of binary factor analysis (BFA) or latent trait model, which has also been widely studied in several fields [114], [10], [110], [108], [105]. Since $y' = Fy$ for any $F \neq I$ will let y' go beyond binary vector, the rotation indeterminacy and scaling indeterminacy have been removed. Moreover, in addition to the constraint $S = A \Lambda A^T + \Sigma_e$ with $\Lambda = diag[q_1(1-q_1), \cdots, q_m(1-q_m)]$, we can also get the constraint equations by considering higher order statistics on both the sides of eq.(6), such that the additive indeterminacy can also be removed, especially when $\Sigma_e = \sigma_e^2 I$. As shown in Fig.16(c), the BFA structure describes samples of x via a mixture of Gaussian densities with their centers located on the vertices of a projected polyhedra. Details are referred to [104].

The mapping $x_t \to y_t$ in the cases of NFA and BFA is implemented by an nonlinear optimization that has to be solved iteratively, which actually wastes the major part of computing cost because each mapping $x_t \to y_t$ has to be performed per sample via an nonlinear optimization. To save the computing cost and also to directly implement $x_t \to y_t$, we ask whether it is able to also learn a parametric model $f(x)$. It has been shown in [107] that a differentiable sigmoid function

$$f(x) = s(Wx + c), s(y) = [s(y^{(1)}), \cdots, s(y^{(m)})]^T. \quad (57)$$

is a reasonable choice and that a sigmoid nonlinearity is a necessary condition for surpassing noise in observation during mapping x into independent components.

With $f(x)$ by eq.(57) in the bi-directional version of BFA shown in Fig.19(c), we are lead to

$$\min_{A,W,\mu,c} \frac{1}{dN} \sum_{t=1}^{N} \|x_t - As(Wx_t + c) - \mu\|^2, \quad (58)$$

which is referred as auto-association learning via three layer net and can be trained in the same way as training a three layer net by the Back-propagation technique [84].

It was experimentally demonstrated that the sigmoid layer $s(Wx_t + c)$ makes feature extraction [16], though by that time it was not noticed that it is related to independent factor analysis.

Particularly, for a $W = A^T$ eq.(58) becomes

$$\min_W \frac{1}{dN} \sum_{t=1}^N \|x_t - W^T s(W x_t)\|^2, \quad (59)$$

which is actually the LMSER learning that was proposed in 1991 as an nonlinear extension of PCA [139]. Though the role of $y = s(Wx)$ was originally interpreted as extracting features, it has been further experimentally demonstrated that the LMSER learning performs ICA with its performance superior to a nonlinear Hebbian learning [54]. Also, it has been successfully used on implementing a binary ICA with noise [146]. Here we see that both LMSER and auto-association learning attempt to perform both a specific ICA that maps x into a uniform density and a specific NFA that fits x via independent factors from a uniform density, with observation noise in consideration. It thus also confirmed the above mentioned experimental observation in [54] and uncovered the reason why LMSER learning is more robust to noise than the nonlinear Hebbian learning does.

Furthermore, the least square learning on a conventional three layer forward networks can also be revisited via modifying the auto-association learning by eq.(58).

From the perspective of BYY system in Fig.32 with $q(x|y) = G(x|Ay, \Sigma_e)$ and $q(y)$ by eq.(4) and eq.(8), extensions of BFA, auto-association, LMSER, and three layer net have been obtained from the BYY harmony learning by eq.(18), with not only criteria for selecting m and adaptive learning algorithms for making learning with m determined automatically. Details are further referred to [104], [107], [105]. Moreover, temporal extension of BFA, LMSER, and three layer net have also been developed into variants of hidden Markov models [100].

V. Concluding Remarks

Fundamentals of statistical learning for knowledge discovery and problem solving have been discussed. Typical dependence structures for Challenge one and typical learning theories for Challenge two have been both surveyed. Moreover, fundamentals and advantages of the BYY harmony learning have also been introduced. We observe that the BYY system provides a unified framework for various dependence structures and the BYY harmony learning provides a promising tool for solving Challenge two for learning on a finite size of samples with model selection made automatically during learning.

As discussed in Sec.II, dependence structures vary from simple ones to more sophisticated ones, along three directions as shown in Fig.57.

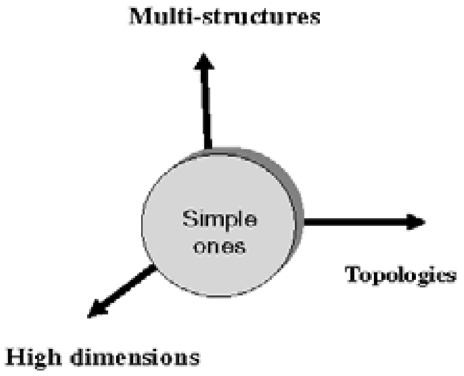


Fig. 57. Three directions

One is from a low dimension space to a high dimension space in order to extend the scope of the world of our observation. To effectively describing dependence structures in a high dimension space, one major solution consists of various independence subspaces. Summarized in Sec.IV are advances obtained from BYY harmony learning on typical independence subspaces.

One is from considering one body only to considering multi-bodies. Ignoring any qualitative relation, many studies and applications have been made in the fields of statistical learning and pattern recognition via the *mixture structures* by eq.(9). These structures can also be viewed from the BYY system perspective, in either a backward architecture as shown in Fig.17 or a bi-directional architecture as shown in Fig.20. In the past 15 years, in help of BYY harmony learning, a number of advances have been obtained on mixture structures and extensions, which are summarized as follows:

- **Mixture of density-structures** One of widely studied case is Gaussian mixture with eq.(10), with the following typical results :
  - The ML learning on Gaussian mixture is implemented by the well known EM algorithm. Before the early 90's, there was a widespread misunderstanding on the EM algorithm. In [141], [129], the first mathematical link between the EM algorithm and the gradient based algorithms has been set up, with three major advantages clearly stated. Thus, this widespread misunderstanding is clarified, which acts as a major stimulate that is responsible for the popularity of the EM algorithm in the neural networks literature in the past decade. Also, readers are referred to Sec.22.9.1(a) and Sec.22.9.2(1) in [102] for a summary on a number of further results on the EM algorithm.
  - The least mean square error (MSE) clustering, as shown in Fig.33, can be regarded as a special case of a ML learning on a Gaussian mixture with each $\Sigma_e = \sigma_e^2 I$. This MSE clustering is implemented by the well known K-MEANS algorithm or its various adaptive variants under

the name of competitive learning. However, they have to be implemented with a pre-specified k. Firstly proposed in [140], [137], rival penalized competitive learning (RPCL) opened a new era of competitive learning studies, with an appropriate number k automatically determined. Moreover, not only RPCL learning has been extended to the so called ellipse clustering with $\Sigma_e \neq \sigma_e^2 I$, but also its link to BYY harmony learning has be set up from several aspects [117], [111], [108]. Further results are referred to Sec.22.9.1(a) and Sec.22.9.2(1)(a)in [102] as well as Sec.23.7.3 in [103].

  - Using the BYY harmony learning on various cases of Gaussian mixture [111], [108], [106], not only criteria have been derived for selecting k in a two stage implementation, but also adaptive algorithms have been developed such that an appropriate k can be determined automatically during learning. Readers are referred to Sec.22.9.1(a) and Sec.23.7.3 in [103] for a summary.
  - Jointly considering various independence spaces in Sec.IV and the mixture structures by eq.(9), we have also get mixtures of various independent subspaces, e.g., as shown in Fig.28. Readers are referred to Sec.22.9.1(b) in [102].

- **Mixture of shape-structures** In many real problems, especially in the computer vision field, as shown in Fig.28, each individual structure is featured by a shape. The task of finding multiple shapes is usually called object detection, for which we have the following results:
  - This type of tasks has been tackled by the well known Hough transform [43], [49]. However, its computing cost is huge while the performance accuracy is rather low. In [145], a new approach was proposed under the name of randomized Hough transform (RHT) that replaces the diverging mapping of Hough transform with a new combined mechanism of random sampling and converging mapping, with significantly improved performances[138], [52]. After developments over a decade, it has already become an important research branch of Hough transform studies. Readers are further referred to [98].
  - The performances of Hough transform degenerate significantly under strong noise, partially observable objects, and a large amount of objects. In [135], [131], a multi-set modelling method has been proposed, with high success detection rate in situations of strong noise, partially observable objects, and a large amount of objects. Readers are referred to [106], [61], [62] for further results.
  - In [98], a unified problem solving paradigm has been developed to combine the advantages of RHT based techniques, learning based approaches, and evidence combination.

- **Combination of multiple decision and inference** Started from the research area of handwritten character recognition at the end of 80's, the problem of multiple classifier combination got an ever increasing attention. Also, in the neural networks literature at the beginning of 90's, topics of combining multiple decisions or inferences have also attracted an ever increasing attention. Listed below are typical results obtained:
  - In [143], an early systematic study has been made on several solving methods for multiple classifier combination as well as its applications in character recognition, which is now widely cited.
  - In [136], a number of results have been obtained on statistical consistency, convergence rates and receptive field size for RBF nets that were among earliest major results obtained in the literature of RBF nets.
  - An alternative model of mixture of experts has been proposed, featured by being easily implemented by the EM algorithm [133], which is also further applied to replace the existing suboptimal two stage algorithm for RBF nets learning [115]. The number of basis functions are determined via either BYY harmony learning or rival penalized competitive learning. Readers are referred to Sec.22.9.1(d) in [102] for a brief summary.

The third direction in Fig.57 is from being ignorance of to taking in consideration of topological relations among multi-bodies. As discussed in Sec.II-C, the simplest and also most widely studied one is a directional linear topology, or equivalently temporal relation. Results on temporal extensions of independent subspaces have also been discussed in Sec. IV. Readers are referred to Sec.22.9.1(f) in [102] for a further summary. Another important topology relation is topological map, which has already discussed in Sec.II-C.2.

ACKNOWLEDGMENT

The work described in this paper was fully supported by a grant from the Research Grant Council of the Hong Kong SAR (project No: CUHK4225/04E).

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] Amari, S., Chen, TP., & Cichocki, A. (1997), "Stability analysis of adaptive blind source separation", *Neural Networks, 10*, 1345-1351.

[5] Amari, SI, Cichocki, A, & Yang, HH (1996), "A new learning algorithm for blind separation of sources", in DS Touretzky, et al, eds, *Advances in Neural Information Processing 8*, MIT Press, 757-763.

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

[7] Attias, H. (1999), "Independent factor analysis", *Neural Computation, 11*, 803-851.

[8] Baldi, P, & Brunak, S (2001), *Bioinformatics: The Machine Learning Approach*, The MIT Press, Cambridge, MA.

[9] Baldi, P. & Hornik, K.(1991). "Back-propagation and unsupervised learning in linear networks,", in Y. Chauvin and D. E. Rumelhart (Eds.), *Back Propagation: Theory, Architectures and Applications*. Erlbaum Associates, 1991.

[10] Bartholomew, DJ & Knott, M., (1999). "Latent variable models and factor analysis", *Kendall's Library of Satistics, Vol. 7*, Oxford University Press, New York, 1999.

[11] Belkin, M. and Niyogi, P., (2003), " Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation,15(6)*, pp1373-1396.

[12] Bell, A & Sejnowski, T (1995), "An information maximization approach to blind separation and blind deconvolution", *Neural Computation, 17*, 1129-1159.

[13] Belouchrani, A., & Cardoso, JF, (1995), "Maximum likelihood source separation by the expectation-maximization technique: deterministic and stochastic implementation", *Proc. NOLTA95*, 49-53.

[14] Bishop, C., Svensen, M. and Williams, C. (1998) "GTM: The generative topographic mapping", *Neural Computation,10*, pp215-234.

[15] Bollerslev,T.(1986), "Generalized autoregressive conditional heteroskedasticity", Journal of Econometrics, 31, 307-327.

[16] Bourlard, H & Kamp, Y (1988), "Auto-association by multilayer Perceptrons and singular value decomposition", Biol. Cyb. 59, 291-294.

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

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

[19] Brockett, RW (1991), "Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems", *Linear Algebra and Its Applications*, 146, 79-91.

[20] Carpenter, GA & Grossberg, S., (1987), "A massively parallel architecture for a self-organizing neural pattern recognition machine", *Computer Vision, Graphics, and Image Processing*, Vol.37, 54-115.

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

[22] Chan, KY, Chu, WS, & Xu, L (2003), "Experimental Comparison between two computational strategies for topological self-organization", *Proc. of IDEAL03*, Lecture Notes in Computer Science, LNCS 2690, Springer-Verlag, pp410-414.

[23] Chen, TP, & Amari, S. (2001), "Unified Stabilization Approach to principal and minor components extraction algorithm," *Neural Networks, 14*, 1377-1387.

[24] Cheung, CC & Xu, L. (2000), "Some Global and Local Convergence Analysis on The Information-Theoretic Independent Component Analysis Approach", *Neurocomputing, 30*, 79-102.

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

[26] Comon, P (1994), "Independent component analysis - a new concept?", *Signal Processing 36*, 287-314.

[27] Cox,T. and Cox, M. (1994), *Multidimensional Scaling*, Chapman&Hall, London.

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

[29] Dempster, AP, Laird, NM, & Rubin, DB, (1977), "Maximum-likelihood from incomplete data via the EM algorithm", *J. of Royal Statistical Society*, B39, 1-38.

[30] Devijver, PA, & Kittler, J (1982), *Pattern Recognition: A Statistical Approach*, Prentice-Hall.

[31] Devroye, L, et al (1996), *A Probability Theory of Pattern Recognition*, Springer.

[32] DiCiccio, TJ, et al, (1997), " Computing Bayes factors by combining simulations and asymptotic Approximations", *Journal of the American Statistical Association*, 92 (439), 903-915.

[33] Duda, RO, & Hart, PE (1973), *Pattern classification and Scene analysis*, Wiley.

[34] Efron, B & Tibshirani, R (1993), *An Introduction to the Bootstrap*, Chaoman and Hall, New York .

[35] Everson, R., & Roberts, S. (1999), " Independent component analysis: A fexible nonlinearity and decorrelating manifold approach", *Neural Computation, 11*, pp1957-1983.

[36] Gaeta, M, & Lacounme, JL (1990), "Source Separation without a priori knowledge: the maximum likelihood solution", in *Proc. EUSIPCO90*, 621-624.

[37] Girolami, M. (1998), "An alternative perspective on adaptive independent component analysis algorithms", *Neural Computation, 10*, 2103-2114.

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

[39] Han, J, & Kamber, M (2001), *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2001.

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

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

[42] Hotelling, H (1936), "Simplified calculation of principal components", *Psychometrika 1*, 27-35.

[43] Hough, PVC, (1962) " Method and means for recognizing complex patterns, *U.S. Patent 3069654*, Dec.18, 1962.

[44] Hu, XL & Xu, L (2005), "An Improved BYY Data Smoothing Learning Criterion for Selecting Subspace Dimension", in preparation.

[45] Hu, XL & Xu, L (2004), "A Comparative Investigation on Subspace Dimension Determination", *Neural Networks*, Vol. 17, (2004), 1051–1059.

[46] Hu, XL & Xu, L (2004), "Investigation on Several Model Selection Criteria for Determining the Number of Cluster", *Neural Information Processing - Letters and Reviews*, Vol.4, No.1, July 2004, pp. 1-10.

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

[48] Hyvarinen, A., Karhunen, J., & Oja, A. (2001), Independent Component Analysis, John Wileys & Sons, Inc., New York.

[49] Illingworth, J. & Kittler, J., (1988), "A survey of the Hough Transform", *Comput. Vision Graphics and Image Process.* 43, 221-238.

[50] Jutten, C & Herault, J (1988), "Independent Component Analysis versus Principal Component Analysis", *Proc. EUSIPCO88*, 643-646.

[51] Kalman, RE, "A New Approach to linear filtering and prediction problems", *Trans. ASME-J Basic Engineering*, March, 35-45, 1960.

[52] Kälviäinen, H., Hirvonen, P., Xu, L., & Oja, E. (1995), "Probabilistic and Non-probabilistic Hough Transforms: Overview and Comparisons", *Image and Vision Computing*, Vol.5, No. 4, pp.239-252.

[53] Karhunen, J.(1982), "On the recursive estimation of the eigenvectors of correlation type matrices", *Lic. Tech. Thesis*,, Helsinki University of Technology.

[54] Karhunen, J. & Joutsensalo, J. (1994), "Representation and separation of signals using nonlinear PCA type learning", *Neural Networks, 7(1)*, 113-127.

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

[56] Kawamoto, M. (2002), "Cerebellum and Motor Cobtrol", *The Handbook of Brain Theory and Neural Networks*, Second edition, (MA Arbib, Ed.), Cambridge, MA: The MIT Press, pp190-195.

[57] Kohonen, T (1982), "Self-organized formation of topologically correct feature maps", *Biological Cybernetics 43*, 59-69.

[58] Kohonen, T (1995), *Self-Organizing Maps* , Springer-Verlag, Berlin.

[59] Lee, T., Girolami, M., & Sejnowski, T. (1999). "Independent component analysis using an extended infomax algorithm for mixed sub-Gaussian and super-Gaussian sources". *Neural Computation*, *11*, 409-433.

[60] Liu, ZY, Chiu, KC, and Xu, L (2004), "The One-bit-Matching Conjecture for Independent Component Analysis", *Neural Computation*, Vol. 16, No. 2, pp. 383-399.

[61] Liu, ZY, Chiu, KC, and Xu, L (2003), " Strip Line Detection and Thinning by RPCL-Based Local PCA", *Pattern Recognition Letters 24*, pp2335-2344, 2003.

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

[63] Lu, W., & Rajapakse, J. (2000), "A neural network for under complete Independent Component Analysis", In *Proc. 8th European Symposium on Artificial Neural Networks*, Bruges, Belgium.

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

[65] Mackey, D (1992a) "A practical Bayesian framework for backpropagation", *Neural Computation*, 4, 448-472.

[66] Mackey, D (1992b) "Bayesian Interpolation", *Neural Computation*, 4, 405-447.

[67] von der Malsburg, Ch (1973), "Self-organization of orientation sensitive cells in the striate cortex", Kybernetik 14, 85-100.

[68] Moulines, E., Cardoso, J., & Gassiat, E. (1997), "Maximum likelihood for blind separation and deconvolution of noisy signals using mixture models", *Proc. ICASSP97*, 3617-3620.

[69] McDonald, R (1985), *Factor Analysis and Related Techniques*, Lawrence Erlbaum.

[70] McLachlan, GJ & Krishnan, T (1997) *The EM Algorithm and Extensions*, John Wiley & Son, INC.

[71] 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.

[72] Oja, E. (1982), "A simplified neuron model as a principal component analyzer", *Journal of Mathematical Biology, 16*, *267-273*.

[73] Oja, E. (1989). Neural networks, principal components, and subspaces", *International Journal of Neural Systems, 1*, 61-68.

[74] Pearlmutter, B., & Parra, L. (1996). "A context-sensitive generalization of ICA", *Proc. ICONIP96*, 151-157.

[75] Pham, DT, Garrat, P., & Jutten, C. (1992), "Separation of a mixture of independent sources through a maximum likelihood approach", *Proc. EUSIPCO92*, 771-774.

[76] Rabiner, L & Juang, BH (1993), *Fundamentals of Speech Recognition*, Prentice Hall, Inc.

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

[78] Ridder, DD, Duin, D., & Kittler, R (2002), " Texture description by independent components", in *Proc. the Joint International workshop on syntactical and structural pattern recognition*, Windsor, Canada.

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

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

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

[82] Roweis, S. and Saul, LK, (2000), "Nonlinear Dimensionality Reduction by Locally Linear Embedding", *Science, 290*, 22 December 2000, pp2323-2326.

[83] Rubi, D & Thayer, D (1976), "EM algorithm for ML factor analysis", *Psychometrika 57*, 69-76.

[84] Rumelhart, DE, Hinton, GE, & Williams, RJ (1986), "Learning internal representations by error propagation", *Parallel Distributed Processing*, **1**, MIT press.

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

[86] 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.

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

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

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

[90] Tenenbaum, JB, Silva, J, and Langford, JC, (2000), "A Global Geometric Framework for Nonlinear Dimensionality Reduction", *Science, 290*, 22 December 2000, pp2319-2323.

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

[92] Tong, L, Inouye, Y, & Liu, R (1993) "Waveform-preserving blind estimation of multiple independent sources", *IEEE Trans. on Signal Processing 41*, 2461-2470.

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

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

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

[96] Welling, M., & Weber, M. (2001), "A constrained EM algorithm for independent component analysis", *Neural Computation, 13, 677-*689.

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

[98] Xu, L. (2005), "A5 Problem Solving Paradigm: A Unified Perspective and New Results on RHT Computing, Mixture Based Learning, and Evidence Combination", invited and featured talk, Proc. of IEEE International Conference on Granular Computing, Beijing, 25-27 July 2005, pp70-77.

[99] Xu, L. (2005),"One-Bit-Matching ICA Theorem, Convex-Concave Programming, and Combinatorial Optimization", *Lecture Notes in Computer Science, Vol 3496: Advances in Neural Networks*, A plenary talk on Second International Symposium on Neural Networks, May 30 - June 1, 2005, Springer-Verlag, pp5-20.

[100] Xu, L. (2004), "Temporal BYY Encoding, Markovian State Spaces, and Space Dimension Determination", *IEEE Trans on Neural Networks, Vol. 15, No. 5*, pp1276-1295.

[101] Xu, L (2004), "Advances on BYY harmony learning: information theoretic perspective, generalized projection geometry, and independent factor auto-determination", *IEEE Trans on Neural Networks, Vol. 15, No. 4*, pp885-902.

[102] 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, pp615-659.

[103] 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, pp661-706.

[104] 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, pp5-18.

[105] Xu, L. (2003), "BYY Learning, Regularized Implementation, and Model Selection on Modular Networks with One Hidden Layer of Binary Units", *Neurocomputing*, Vol.51, p227-301.

[106] Xu, L. (2003), "Data smoothing regularization, multi-sets-learning, and problem solving strategies", *Neural Networks, Vol. 15, Nos. 5-6*, 817-825.

[107] 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.

[108] Xu, L (2002), "BYY Harmony Learning, Structural RPCL, and Topological Self-Organizing on Mixture Models ", *Neural Networks, Vol. 15, Nos. 8-9*, 1125-1151.

[109] 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, pp1231-1237.

[110] Xu, L (2001), "BYY Harmony Learning, Independent State Space and Generalized APT Financial Analyses ", *IEEE Trans on Neural Networks*, **12** (4), 822-849.

[111] 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.

[112] Xu, L (2001), "An Overview on Unsupervised Learning from Data Mining Perspective", *Advances in Self-Organizing Maps*, Nigel Allison, et al eds, Springer-Verlag, June, pp181-210 (2001).

[113] Xu, L (2000), "Temporal BYY Learning for State Space Approach, Hidden Markov Model and Blind Source Separation", *IEEE Trans on Signal Processing 48*, 2132-2144.

[114] Xu, L (1998), "Bayesian Kullback Ying-Yang Dependence Reduction Theory", *Neurocomputing 22 (1-3)*, 81-112, 1998.

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

[116] Xu, L(1998), "Bayesian Ying-Yang Learning Theory For Data Dimension Reduction and Determination ", *Journal of Computational Intelligence in Finance*, Finance & Technology Publishing, Vol.6, No.5, pp 6-18, 1998.

[117] Xu, L, (1998), Rival Penalized Competitive Learning, Finite Mixture, and Multisets Clustering, *Proc. of IJCNN98*, May 5-9, 1998, Anchorage, Alaska, Vol.II, pp2525-2530.

[118] Xu, L, Cheung, CC, & Amari, SI (1998) "Learned Parametric Mixture Based ICA Algorithm", *Neurocomputing 22 (1-3)*, 69-80.

[119] Xu, L, Cheung, CC, & Amari, SI (1998), "Further Results on Nonlinearity and Separation Capability of A Linear Mixture ICA Method and Learned Parametric Mixture Algorithm", *Proc. I&ANN'98*, Feb. 9-10, 1998, Tenerife, Spain, pp39-44.

[120] Xu, L (1997),"Bayesian Ying-Yang System and Theory as A Unified Statistical Learning Approach: (I) Unsupervised and Semi-Unsupervised Learning", in S Amari and N Kassabov eds, *Brain-like Computing and Intelligent Information Systems*, Springer-Verlag, pp241-274.

[121] Xu, L (1997), "Bayesian Ying-Yang System and Theory as A Unified Statistical Learning Approach (II): From Unsupervised Learning to Supervised Learning and Temporal Modeling", in KM. Wong, et al, eds, *Theoretical Aspects of Neural Computation: A Multidisciplinary Perspective*, Springer-Verlag, pp25-42.

[122] Xu, L (1997), "Bayesian Ying-Yang System and Theory as A Unified Statistical Learning Approach (III): Models and Algorithms for Dependence Reduction, Data Dimension Reduction, ICA and Supervised Learning", in KM. Wong, et al, eds, *Theoretical Aspects of Neural Computation: A Multidisciplinary Perspective*, Springer-Verlag, pp43-60.

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

[124] Xu, L, Cheung, CC, Yang, HH, & Amari, SI (1997), "Independent Component Analysis by The Information-Theoretic Approach with Mixture of Density", *Proc. IJCNN97*, Vol. III, 1821-1826.

[125] Xu, L, Cheung, CC, Ruan, J, & Amari, SI (1997), "Nonlinearity and Separation Capability: Further Justification for the ICA Algorithm with A Learned Mixture of Parametric Densities", *Proc. ESANN97*, Bruges, April 16-18, 1997, pp291-296.

[126] Xu, L (1997), "Bayesian Ying-Yang Learning Based ICA Models", *Proc. 1997 IEEE Signal Processing Society Workshop on Neural Networks for Signal Processing VII*, Sept. 24-26, 1997, Florida, pp476-485.

[127] Xu, L. Yang, HH, & Amari, SI (1996), "Signal Source Separation by Mixtures Accumulative Distribution Functions or Mixture of Bell-Shape Density Distribution Functions", Research Proposal, presented at *FRONTIER FORUM (speakers: D. Sherrington, S. Tanaka, L.Xu & J. F. Cardoso)*, organized by S.Amari, S.Tanaka & A.Cichocki, RIKEN, Japan, April 10, 1996.

[128] Xu, L, (1996), "A Unified Learning Scheme: Bayesian-Kullback YING-YANG Machine", *Advances in Neural Information Processing Systems*, 8, 444-450.

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

[130] Xu, L, (1995), "Bayesian-Kullback Coupled YING-YANG Machines: Unified Learnings and New Results on Vector Quantization", *Proc. Intl. Conf. on Neural Information Processing (ICONIP96)*, Oct 30-Nov.3, 1995, Beijing, China, pp977-988.

[131] Xu, L (1995), "A unified learning framework: multisets modeling learning," *Proceedings of 1995 World Congress on Neural Networks*, vol.1, pp35-42.

[132] Xu, L, (1995), "Advances on Three Streams of PCA Studies", Invited Talk, *Proc. of 1995 IEEE Intl. Conf. on Neural Networks and Signal Processing*, Dec. 10-13, 1995, Nanjing, Vol.I, pp480-483.

[133] 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.

[134] Xu, L, & Yuille, AL (1995), "Robust Principal Component Analysis by Self-Organizing Rules Based on Statistical Physics Approach", *IEEE Trans. on Neural Networks*, Vol.6, No.1, 131-143.

[135] Xu, L, (1994), "Multisets Modeling Learning: An Unified Theory for Supervised and Unsupervised Learning", Invited Talk, *Proc. IEEE ICNN94*, June 26-July 2, 1994, Orlando, Florida, Vol.I, pp.315-320.

[136] 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.

[137] 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.

[138] 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.

[139] Xu, L (1991&93) "Least mean square error reconstruction for self-organizing neural-nets", *Neural Networks 6, 627*-648, 1993. Its early version on *Proc. IJCNN91'Singapore*, 2363-2373, 1991.

[140] 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)*, Aug.30–Sept.3, 1992, Hauge, Netherlands, Vol.I, pp.*672-675*.

[141] Xu, L., & Jordan, M. (1992), "Thoeretical and experimental studies of the EM algorithm fo unsupervised learning based on finite Gaussin mixture", MIT Computational Cognitive Science, Tech. Rep. 9302, Dept. of Brain and Cognitive Science, MIT Cambridge, MA.

[142] Xu, L., Oja, E., & Suen, C.Y. (1992), "Modified Hebbian learning for curve and surface fitting", *Neural Networks, 5*, pp393-407.

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

[144] Xu, L., Krzyzak, A., & Oja, E. (1991), "A Neural Net for Dual Subspace Pattern Recognition Methods", *Intl J. of Neural Systems, 2(3)*, 169-184.

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

[146] Zhang, B. L., Xu, L., & Fu, M. Y. (1996), "Learning Multiple Causes by Competition Enhanced Least Mean Square Error Reconstruction", *Intl J. of Neural Systems, 7(3)*, 223-236.

Most of the above listed papers authored or co-authored by Xu, L. can be obtained via WWW site:

http://www.cse.cuhk.edu.hk/~lxu/