PERGAMON

AVAILABLE AT
www.ComputerScienceWeb.com
POWERED BY SCIENCE DIRECT®

Neural Networks 16 (2003) 817–825

Neural Networks

www.elsevier.com/locate/neunet

2003 Special issue

Data smoothing regularization, multi-sets-learning, and problem solving strategies

Lei Xu*

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

Abstract

First, we briefly introduce the basic idea of data smoothing regularization, which was firstly proposed by Xu [Brain-like computing and intelligent information systems (1997) 241] for parameter learning in a way similar to Tikhonov regularization but with an easy solution to the difficulty of determining an appropriate hyper-parameter. Also, the roles of this regularization are demonstrated on Gaussian-mixture via smoothed versions of the EM algorithm, the BYY model selection criterion, adaptive harmony algorithm as well as its related Rival penalized competitive learning. Second, these studies are extended to a mixture of reconstruction errors of Gaussian types, which provides a new probabilistic formulation for the multi-sets learning approach [Proc. IEEE ICNN94 I (1994) 315] that learns multiple objects in typical geometrical structures such as points, lines, hyperplanes, circles, ellipses, and templates of given shapes. Finally, insights are provided on three problem solving strategies, namely the competition-penalty adaptation based learning, the global evidence accumulation based selection, and the guess-test based decision, with a general problem solving paradigm suggested.

Keywords: Data smoothing; Tikhonov regularization; Gaussian mixture; Multiple objects; Problem solving

1. Introduction

It is well understood that regularization is of key important for parametric modeling or neural networks learning on a finite set of samples. Several regularization techniques have been studied in the literature. They are closely related to the well known Tikhonov regularization (Tikhonov & Arsenin, 1977; Girosi, 1995), featured by adding to the fitting error with a regularizing term that is weighted by a so called hyper-parameter. Conceptually, this hyper-parameter can be further determined via one of several methods, including estimating generalization error bound, cross validation, Bayesian approach, and minimum description length, etc. In implementation, however, they not only suffer extensive computational cost but also are able to provide a rough solution only.

It is also well known for several decades, e.g. in the literatures of signal processing or control theory, that adding a noise with an appropriate variance to a finite set of samples will help parameter estimation or system modeling. In the literature of neural networks, it has been shown that training with noise is equivalent to Tikhonov regularization (Bishop, 1995).

When the size of noise samples are infinite or large enough, it follows from probability theory that adding noise to samples is equivalent to the convolution of the empirical density obtained directly from samples with a smoothing kernel function. This nature is directly used in a non-parametric density estimation under the name Parzen window estimation. When the kernel is a Gaussian function, we have

$$p_h(u) = \frac{1}{N}\sum_{t=1}^{N} G(u|u_t, \Sigma_h), \qquad \Sigma_h = h^2 I, \tag{1}$$

where and throughout this paper, $G(u|m, \Sigma)$ denotes a Gaussian density with a mean vector m and covariance matrix Σ. Particularly, $p_h(u)$ returns back to the empirical density when $h = 0$. That is,

$$p_0(u) = \frac{1}{N}\sum_{t=1}^{N} \delta(u - u_t). \tag{2}$$

Therefore, this h is usually called smoothing parameter and takes a role similar to the above noise variance and the hyper-parameter in Tikhonov regularization. Though

* Tel.: +852-2609-8423; fax: +852-2603-5024.
E-mail address: lxu@cse.cuhk.edu.hk (L. Xu).

0893-6080/03/$ - see front matter © 2003 Elsevier Science Ltd. All rights reserved.
doi:10.1016/S0893-6080(03)00119-9

many studies have been made theoretically on how to estimating an appropriate smoothing parameter (Devroye et al., 1996), in implementation they share difficulties similar to that discussed for determining a hyper-parameter.

The idea of data smoothing regularization came firstly in (Xu, 1997a–c) from using the Parzen window estimator by Eq. (1) in place of directly using empirical density by Eq. (2) on parameter learning of the Bayesian Ying Yang (BYY) system such that the effect of a finite size of samples is regularized via the smoothing role of an appropriate smoothing parameter h that is decided during learning. The BYY harmony learning was firstly proposed in 1995 (Xu, 1995a, 1996b) and systematically developed in the past several years (Xu, 2000a,b, 2001a,b, 2002, 2003). This BYY harmony learning acts as a general statistical learning framework such that not only a number of existing major learning problems and learning methods are revisited as special cases from a unified perspective, but also a harmony learning theory is developed with a new learning mechanism that makes model selection implemented either *automatically* during parameter learning or *subsequently after* parameter learning via a new class of model selection criteria obtained from this mechanism, with new insights and a series of new results. Further details are referred to Xu (2002, 2003).

After briefly introducing the basic idea of data smoothing regularization in Section 2, this paper will focus on its role in two types of finite mixture models via both the maximum likelihood (ML) learning and the BYY harmony learning. In Section 3, the roles of data smoothing regularization are demonstrated on Gaussian-mixture via smoothed versions of the EM algorithm, the BYY model selection criterion, adaptive harmony learning as well as its related Rival Penalized Competitive Learning (RPCL). In Section 4, these studies are extended to a mixture of reconstruction errors of Gaussian types as a new probabilistic formulation for the multi-sets learning approach (Xu, 1994) that learns multiple objects via typical geometrical structures such as points, lines, hyperplanes, circles, ellipses, and templates of given shapes. Moreover, insights are provided in Section 5 on three problem solving strategies, namely the competition-penalty adaptation based learning, the global evidence accumulation based selection, and the guess-test based decision, with a general problem solving paradigm suggested.

2. Data smoothing regularization

2.1. Data smoothing regularization

Given a parametric model $q(\mu|\theta)$, learning with data smoothing regularization is made via maximizing the following criterion (Xu, 1998a, 1999, 2002, 2003):

$$L_D(\theta, h) = L_h(\theta) + Z(h), \tag{3}$$

where $L_h(\theta)$ is a smoothed version of the likelihood function $L_0(\theta)$ as follows:

$$\begin{aligned} L_h(\theta) &= \int p_h(u) \ln q(u|\theta) \mathrm{d}u, \\ L_0(\theta) &= \int p_0(u) \ln q(u|\theta) \mathrm{d}u = \frac{1}{N} \sum_{t=1}^{N} \ln q(u_t|\theta), \end{aligned} \tag{4}$$

with the empirical density $p_0(u)$ replaced by $p_h(u)$. We have the following general form (Xu, 1999, 2002, 2003):

$$\begin{aligned} L_h(\theta) &\approx L_0(\theta) - 0.5 R_h(\theta), \\ R_h(\theta) &= -\frac{1}{N} \sum_t Tr\left[\sum\nolimits_h \frac{\partial^2 \ln q(u|\theta)}{\partial u \, \partial u^{\mathrm{T}}} \right]_{u=u_t}. \end{aligned} \tag{5}$$

It follows from $\Sigma_h = h^2 I$ that

$$R_h(\theta) = h^2 \pi_q, \pi_q = -\frac{1}{N} \sum_t Tr\left[\frac{\partial^2 \ln q(u|\theta)}{\partial u \, \partial u^{\mathrm{T}}} \right]_{u=u_t}. \tag{6}$$

which provides a Tikhonov-type regularization (Tikhonov & Arsenin, 1977; Girosi, 1995) on $L_h(\theta)$ and the role h is equivalent to the hyper-parameter in (Bishop, 1995).

The new thing in Eq. (3) is that h can be decided by $\max_{\theta,h} L_h(\theta)$ due to the role of $Z(h)$, which can be one of two choices (Xu, 1998a, 1999, 2002, 2003):

$$Z(h) = \begin{cases} -\int p_h(u) \ln p_h(u) \mathrm{d}u, & \text{(a)}, \\ -\ln \sum_{t=1}^{N} p_h(u_t), & \text{(b)}. \end{cases} \tag{7}$$

In both the cases, it contains a dominated term $R_h(\theta)$ that increases as h increases. On the other hand, it follows from Eqs. (5) and (6) that $L_h(\theta)$ is maximized as $h \to 0$ since π_q is always non-negative. As a result, the trade-off of two aspects results in an appropriate value for h.

The two types of $Z(h)$ lead to two types of data smoothing regularization. It follows from Eq. (5) that

$$Z(h) = 0.5 \ln|\Sigma_h| + G(h) + \ln N + c_z, \quad c_z \text{ is constant},$$

$$G(h) = \begin{cases} \frac{-1}{N} \sum_{\tau=1}^{N} \ln\left[\sum_{t=1}^{N} \mathrm{e}^{-d_h(u_t, u_\tau)} \right], & \text{(a)}, \\ -\ln \sum_{\tau=1}^{N} \left[\sum_{t=1}^{N} \mathrm{e}^{-d_h(u_t, u_\tau)} \right], & \text{(b)}. \end{cases} \tag{8}$$

$$d_h(u_t, u_\tau) = 0.5(u_t - u_\tau)^{\mathrm{T}} \Sigma_h^{-1} (u_t - u_\tau).$$

As observed in Xu (2002), two types of regularization in Eq. (8) tends to being equivalent as the size N of samples increases to be large enough. The case (a) is more suitable when $q(u|\theta)$ is modeled directly via the smoothed likelihood

$L_h(\theta)$ in Eq. (3) but with no consideration on model selection. While the case (b) is more suitable for making the harmony learning with model selection (Xu, 2001a, 2002, 2003) on a Bayesian Ying–Yang system that represents $q(u|\theta)$ as its marginal density.

Moreover, it follows from Eq. (3) that the learning can be implemented by alternatively (Xu, 1998a–c)

$$\begin{array}{ll} \text{(a)} & \text{Estimating } \theta \text{ via } \max_{\theta} L_h(\theta) \text{ with } h \text{ fixed,} \\ \text{(b)} & \text{Estimating } h \text{ via } \max_{h} L_D(\theta, h) \text{ with } \theta \text{ fixed.} \end{array} \tag{9}$$

e.g. the step (b) can be made via the gradient ascent iteration:

$$h^{\text{new}} = h^{\text{old}} + \eta \Delta h, \qquad \Delta h = \frac{\mathrm{d}L_D(\theta, h)}{\mathrm{d}h}. \tag{10}$$

Alternatively, Δh can also simply given by (Xu, 1999)

$$\Delta h = \begin{cases} \eta, & \text{if } L_D(\theta, h+\eta) > L_D(\theta, h), \\ -\eta, & \text{if } L_D(\theta, h-\eta) > L_D(\theta, h), \\ 0, & \text{otherwise.} \end{cases} \tag{11}$$

In the following we further introduce the details of Eq. (9) in two typical situations:

(1) $\Sigma_h = h^2 I$ in Eq. (1). It applies to the cases that all the components in an observed sample vector should be equally smoothed, e.g. the case of estimating a density $q(x|\theta)$ or equivalently a joint density $q(x, z|\theta)$ that becomes the same as the former by $u = [x, z]$. In the case, we have Eq. (6). It further follows from Eqs. (5) and (8), $\max_h L_D(\theta, h)$ can be made via (Xu, 2002)

$$\frac{\mathrm{d}L_D(\theta, h)}{\mathrm{d}h} \approx -h\pi_q + \mathrm{d}/h + \mathrm{d}h_{u,0}^2/h^3,$$

$$h_{u,0}^2 = \frac{1}{d}\sum_{\tau=1}^{N}\sum_{t=1}^{N} p_t, \tau \|u_t - u_\tau\|^2,$$

$$p_{t,\tau} = \begin{cases} \dfrac{1}{N}\dfrac{\mathrm{e}^{-d_h(u_t,u_\tau)}}{\sum_{t=1}^{N}\mathrm{e}^{-d_h(u_t,u_\tau)}}, & \text{(a)}, \\ \dfrac{\mathrm{e}^{-d_h(u_t,u_\tau)}}{\sum_{\tau=1}^{N}\sum_{t=1}^{N}\mathrm{e}^{-d_h(u_t,u_\tau)}}, & \text{(b)}, \end{cases} \tag{12}$$

where d is the dimension of u. Thus, the estimation of h can be made via solving a positive root of $v^2\pi_q + vd + h_0^2 d = 0$ with $v = h^2$ and h_0^2 regarded approximately as a constant. That is,

$$h^2 = \frac{2h_{u,0}^2}{1 + \sqrt{1 + 4h_{u,0}^2 d^{-1}\pi_q}}. \tag{13}$$

which can be either used directly or as an initial value for Eq. (10).

(2) $\Sigma_h = \mathrm{diag}[h_x^2 I_x, h_z^2 I_z]$ in Eq. (1) with h denoting a vector $[h_x, h_z]^{\mathrm{T}}$. It applies to the cases that the components in an observed sample vector can be divided into two parts (Xu, 1999). Each part has a quite different statistical property and thus should be smoothed separately, where I_x, I_z denotes the unit matrix in the space of x, z, respectively. Correspondingly, $q(x, z|\theta) = q(z|x, \theta_{z|x})q(x|\theta_x)$ also has different structures on z, x. In the case, Eq. (6) is replaced with

$$R_h(\theta) = 0.5h_x^2[\pi_{q(z|x)}^x + \pi_{q(x)}^x] + 0.5h_z^2\pi_{q(z|x)}^z,$$

$$\begin{aligned} \pi_{q(x)}^x &= -\frac{1}{N}\sum_t Tr\left[\frac{\partial^2 \ln q(x|\theta_x)}{\partial x\, \partial x^{\mathrm{T}}}\right]_{x=x_t}, \\ \pi_{q(z|x)}^x &= -\frac{1}{N}\sum_t Tr\left[\frac{\partial^2 \ln q(z|x, \theta_{z|x})}{\partial x\, \partial x^{\mathrm{T}}}\right]_{x=x_t, z=z_t}, \\ \pi_{q(z|x)}^z &= -\frac{1}{N}\sum_t Tr\left[\frac{\partial^2 \ln q(z|x, \theta_{z|x})}{\partial_z\, \partial_z^{\mathrm{T}}}\right]_{x=x_t, z=z_t}. \end{aligned} \tag{14}$$

Moreover, Eq. (12) is replaced by

$$\begin{aligned} \frac{\mathrm{d}L_D(\theta, h)}{\mathrm{d}h_x} &\approx -h_x[\pi_{q(z|x)}^x + \pi_{q(x)}^x] + \frac{d_x}{h_x} + \frac{\mathrm{d}h_{x,0}^2}{h_x^3}, \\ \frac{\mathrm{d}L_D(\theta, h)}{\mathrm{d}h_z} &\approx -h_z\pi_{q(z|x)}^z + \frac{d_z}{h_z} + \frac{\mathrm{d}h_{z,0}^2}{h_z^3}. \end{aligned} \tag{15}$$

where d_x, d_z are the dimension of x, z, respectively, and $h_{x,0}^2$, $h_{z,0}^2$ are obtained from $h_{u,0}^2$ with u replaced by x, z, respectively. Thus, the root by Eq. (13) becomes

$$\begin{aligned} h_x^2 &= \frac{2h_{x,0}^2}{1 + \sqrt{1 + 4h_{x,0}^2 d_x^{-1}[\pi_{q(z|x)}^x + \pi_{q(x)}^x]}}, \\ h_z^2 &= \frac{2h_{z,0}^2}{1 + \sqrt{1 + 4h_{z,0}^2 d_z^{-1}\pi_{q(z|x)}^z}}. \end{aligned} \tag{16}$$

The above studies apply to both the case of estimating a joint density $q(z|x, \theta_{z|x})q(x|\theta_x)$ and the case of estimating a conditional density $q(z|x, \theta_{z|x})$ for supervised learning on the regression function $E(z|x) = \int z q(z|x, \theta_{z|x})\mathrm{d}z$. Typical examples of the latter include three layer forward net, RBF nets and mixture of experts. To do so, we can set $q(x|\theta_x)$ to be either of the following two choices:

$$q(x|\theta_x) = \begin{cases} p_{h_x}(x), & \text{(a) by Eq. (1),} \\ p_o(x), & \text{(b) the unknown true density.} \end{cases} \tag{17}$$

Thus, we approximately have

$$\pi_{q(x)}^x = \begin{cases} -d/h_x^2, & \text{(a) by Eq. (1)} \\ -Tr[S^{-1}], & \text{(b) the unknown true density,} \end{cases} \tag{18}$$

where S is the sample covariance matrix estimated directly from samples of x (Xu, 1999). Thus, the maximum

likelihood learning on $q(z|x,\theta_{z|x})$ with data smoothing regularization can be implemented via $\max_{\theta,h} L_h(\theta)$ with $\pi^x_{q(x)}$ substituted into Eqs. (14)–(16).

Being different from that in Eq. (8) of the case (a), an alternative estimate of $Z(h)$ is also given as follows (Xu, 1999, 2000b):

$$Z(h) = 0.5\ln|\Sigma_h| - G(h) - 0.5d\frac{h_0^2 - e_0^2}{h^2},$$
$$e_0^2 = \frac{1}{dN}\sum_{\tau=1}^{N}\left\|u_t - \sum_{t=1}^{N} Np_{t,\tau}u_\tau\right\|^2, \tag{19}$$

where $p_{t,\tau}$ is same as in Eq. (12). We can get both $dL_D(\theta,h)/dh$ in Eq. (12) and h^2 in Eq. (13) simply with h_0^2 replaced by e_0^2. Similarly, we can get $e_{x,0}^2$, $e_{z,o}^2$ with u_t replaced by x_t, z_t, and then get $dL_D(\theta,h)/dh_x$ and $dL_D(\theta,h)/dh_z$ in Eq. (15) and h_x^2, h_z^2 in Eq. (16) with $h_{x,0}^2$, $h_{z,0}^2$ replaced by $e_{x,0}^2$, $e_{z,0}^2$. The difference between $Z(h)$ in Eq. (19) from that of the case (a) in Eq. (7) is whether the second order information within $p_h(u)$ is considered.

2.2. Historic remarks

The data smoothing learning by Eq. (3) with $Z(h)$ by the cases (a) in Eq. (7) came firstly from the implementation of BYY parameter learning via minimizing the Kullback–Leiber (KL) divergence between the Yang machine and Ying machine in the case of the backward architecture (Xu, 1995a, 1996b). With $p_h(x)$ by Eq. (1), it was firstly proposed under the name of data smoothing by Eq. (16) in Xu (1997b) and Eq. (3.10) in Xu (1997a) that an appropriate h is also learned via minimizing the KL divergence, which becomes equivalent to

$$\min_{\theta,h} \mathrm{KL}(\theta,h), \qquad \mathrm{KL}(\theta,h) = \int p_h(x)\ln\frac{p_h(x)}{q(x|\theta)}dx, \tag{20}$$

which was firstly presented by Eq. (7) in Xu (1997c). Obviously, it can be rewritten into Eq. (3) with $Z(h)$ given by the case (a) of Eq. (7). In a BYY system, $q(x|\theta) = \int q(x|y)q(y)dy$ is the marginal density represented by the Ying machine. Generally, being independent of the BYY system, $q(x|\theta)$ can be any parametric model for density estimation. Also in Xu (1997b), the data smoothing regularization is suggested on $q(z|x,\theta_{z|x})$ for supervised learning of three layer forward net and mixture of experts.

A preliminary systematic study on data smoothing regularization was provided in Xu (1998a), including (a) two ways to tackle the integral in $L_h(\theta)$; (b) the alternative strategy of Eq. (9); (c) extensions to estimating $q(z|x,\theta_{z|x})$ for supervised learning with three layer forward net, RBF nets and mixture of experts; (d) the role of h in the model selection by BYY harmony learning. Further progresses in all the four aspects were presented in Xu (1999), including (1) handling the integral in $L_h(\theta)$ in help of

$$\int G(x|x_t,h^2I)F(x)dx \approx F(x_t) + 0.5h^2Tr[H_F], \tag{21}$$

with H_F being the Hessian matrix of $F(x)$; (2) the use of Δh in Eq. (11) and the use of Eq. (21) in handling $-\int p_h(x)\ln p_h(x)dx$; (3) the use of Eq. (1) with $\Sigma_h = \mathrm{diag}[h_x^2I_x, h_z^2I_z]$ for estimating $q(z|x,\theta_{z|x})$; (d) the suggestion of the case (b) in Eq. (18) for learning of three layer forward net, RBF nets and mixture of experts.

A systematic summary can be found in Xu (2000b) on the studies of Eq. (3) with $Z(h)$ by the case (a) in Eq. (7). The data smoothing learning by Eq. (3) with $Z(h)$ by the case (b) in Eq. (7) was firstly proposed in Xu (2001b). The detailed equations as those introduced in Section 2.1 were firstly provided in Xu (2001a) and further discussions in comparison with the case (a) are made in Xu (2002, 2003).

3. Gaussian mixture and multi-sets-mixture

3.1. Gaussian mixture and smoothed EM algorithm

Specifically we start at considering a Gaussian mixture:

$$q(x|\theta) = \sum_{\ell=1}^{k}\alpha_\ell G(x|m_\ell,\Sigma_\ell). \tag{22}$$

Give a known k, as proposed in Xu (1997b), the estimation of can be implemented by the following smoothed EM algorithm:

$$E\text{ step}: p_{\ell,t} = \alpha_\ell G_t(\theta_\ell)/\sum_{\ell=1}^{k}\alpha_\ell G_t(\theta_\ell),$$
$$G_t(\theta_\ell) = G(x_t|m_\ell,\Sigma_\ell), \qquad \theta_\ell = \{m_\ell,\Sigma_\ell\}$$
$$M\text{ step}: \alpha_\ell = \frac{1}{N}\sum_{t=1}^{N}p_{\ell,t}, \qquad m_\ell = \frac{1}{N\alpha_\ell}\sum_{t=1}^{N}p_{\ell,t}x_t, \tag{23}$$
$$e(x_t,m_\ell) = x_t - m_\ell,$$
$$\Sigma_\ell = h^2I + \frac{1}{N\alpha_\ell}\sum_{t=1}^{N}p_{\ell,t}e(x_t,m_l)e^{\mathrm{T}}(x_t,m_\ell),$$

which is different from the EM algorithm (Dempster, Laird & Rubin, 1977; Redner & Walker, 1984) with a smoothing parameter h^2 added to the diagonal elements of Σ_ℓ. Also, it follows from Eq. (5) that

$$\pi_q = -\frac{1}{N}\sum_{t=1}^{N}\sum_{\ell=1}^{k}p_{\ell,t}Tr[\Sigma_\ell^{-1}] = -\sum_{\ell=1}^{k}\alpha_\ell Tr[\Sigma_\ell^{-1}], \tag{24}$$

and the estimation of h^2 is made by Eq. (13) and can be further elaborated via Eq. (10).

When k is unknown, we need also to select an appropriate value for it. The task is called model selection, which is usually made on selecting a best value for k via

$\min_k J(k)$, with $J(k)$ called model selection criterion. In Xu (1997d), a criterion is proposed from the BYY harmony learning, without considering data smoothing regularization. Considering data smoothing regularization in the case of a finite size of samples, this criterion is further extended into (Xu, 2002):

$$J(k) = \sum_{\ell=1}^{k} \alpha_\ell \ln \frac{|\Sigma_\ell|^{0.5}}{\alpha_\ell} + 0.5h^2 \sum_{l=1}^{k} \alpha_\ell Tr[\Sigma_\ell^{-1}], \tag{25}$$

which returns to the one in Xu (1997d) if $h = 0$. As experimentally shown in Hu and Xu (2003), $J(k)$ with data smoothing can select the correct value of k on a data of a small size of samples, while the $J(k)$ by Eq. (25) with $h = 0$ (i.e. without data smoothing) sometimes fail to find the correct value of k. On a data set of enough number of samples, the above $J(k)$ without or with data smoothing can both select the correct value of k. On the same data set of a large size, however, typical existing criteria of the Akaike's information criterion (Akaike, 1974), the Schwarz's Bayesian inference criterion (Schwarz, 1978) or equivalently the minimum description length criterion (Rissanen, 1999) all result in wrong solutions.

Via the BYY harmony learning, k may also be selected in parallel automatically during estimating θ. For the Gaussian mixture by Eq. (22), the resulted algorithm is a modification of Eq. (23) with the E step simplified into the following hard-cut form

$$p_{\ell,t} = \begin{cases} 1, & \ell = \arg\max_j [\alpha_j G_t(\theta_j)], \\ 0, & \text{otherwise} \end{cases}. \tag{26}$$

As long as k is initially a number large enough, the competitive role of Eq. (26) will make α_j, that corresponds to an extra Gaussian, become very small or zero (Xu, 2001b, 2002). Thus, the corresponding Gaussian can be discarded with an appropriate number k determined automatically.

The winner-take-all (WTA) nature of Eq. (26) may cause the learning process to be stuck at a local solution (Xu 2001b), especially on a finite size of samples. This problem is compensated by the regularization of h^2 during the M-step.

The problem caused by this WTA can also be solved by another type of regularization called normalization that introduces a new conscience de-learning mechanism similar to the RPCL (Xu, Krzyzak, & Oja, 1993), which gets

$$c = \arg\min_j d_{j,t}, \qquad r = \arg\min_{j \neq c} d_{j,t},$$

$$d_{j,t} = -\ln[\alpha_j G_t(\theta_j)], \qquad p_{\ell,t} = \begin{cases} 1, & \text{if } \ell = c, \\ -\gamma, & \text{if } \ell = r, \\ 0, & \text{otherwise}, \end{cases} \tag{27}$$

where γ approximately takes a value between [0.1, 0.05]. The first winner will learn while the second winner (or called rival) will be de-learned by a small degree. Though, RPCL learning was originally proposed in a heuristic way. It has been further found that it can be regarded as a simplification or a variant of BYY harmony learning with normalization regularization. The details are referred to Xu (2001b, 2002).

The RPCL learning algorithm, an adaptive version of Eq. (23) for maximum likelihood (ML) learning, and BYY harmony (H)-learning with the E step replaced by Eq. (26) can be unified into the following procedure:

$$(a)\ p_{\ell,t} = \begin{cases} \text{by Eq. (23)}, & \text{ML-Learning}, \\ \text{by Eq. (26)}, & \text{Harmony-Learning}, \\ \text{by Eq. (27)} & \text{RPCL-Learning}, \\ p(\ell|x_t) - \gamma q(\ell|x_t), & \text{A general form}. \end{cases}$$

where $p(\ell|x_t) \geq 0$,

$$\sum_{\ell=1}^{k} p(\ell|x_t) = 1; \quad q(\ell|x_t) \geq 0, \quad \sum_{\ell=1}^{k} q(\ell|x_t) = 1. \tag{28}$$

$$(b)\ \alpha_\ell = \beta_\ell^{2\text{new}} \Big/ \sum_{\ell=1}^{k} \beta_\ell^{2\text{new}},$$

$$\beta_\ell^{\text{new}} = \beta_\ell^{\text{old}} + \eta(p_{\ell,t} - \alpha_\ell^{\text{old}} \sum_{\ell=1}^{k} p_{\ell,t})/\beta_\ell^{\text{old}},$$

$$(c)\ m_\ell^{\text{new}} = m_\ell^{\text{old}} + \eta p_{\ell,t} e_{\ell,t},\ e_{\ell,t} = x_t - m_\ell,$$

$$\Sigma_\ell = S_\ell S_\ell^{\mathrm{T}},\ S_\ell^{\text{new}} = S_\ell^{\text{old}} + \eta p_{\ell,t} G_{\Sigma_\ell}^{\text{old}} S_\ell^{\text{old}},$$

$$G_{\Sigma_\ell} = \Sigma_\ell^{-1} e_{\ell,t} e_{\ell,t}^{\mathrm{T}} \Sigma_\ell^{-1} - \Sigma_\ell^{-1}.$$

where the updating rules on α_ℓ, Σ_ℓ guarantee the satisfaction, even when $p_{\ell,t} < 0$, that $\alpha_\ell \geq 0$, $\sum_{\ell=1}^{k} \times \alpha_\ell = 1$ and Σ_ℓ is non-negative definite. The general form of $p_{\ell,t} = p(\ell|x_t) - \gamma q(\ell|x_t)$ includes the other three cases as its special cases. When $\gamma = 0$, we get the ML-Learning case for $p(\ell|x_t) = p_{\ell,t}$ by Eq. (23) and the Harmony-Learning case for $p(\ell|x_t) = p_{\ell,t}$ by Eq. (26). When γ approximately takes a value between [0.1, 0.05], we get the RPCL-Learning when $p(\ell|x_t) = \bar{\delta}_{\ell,c}$, $q(\ell|x_t) = \bar{\delta}_{\ell,r}$ with c, r by Eq. (27) and

$$\bar{\delta}_{i,j} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$$

The general form also leads to the mentioned BYY harmony learning with normalization regularization when $p(\ell|x_t) = \bar{\delta}_{\ell,c}$ and

$$q(\ell|x_t) = \begin{cases} \dfrac{\alpha_j G(x_i(m_j, \Sigma_j))}{\sum_{\ell \in L_k} \alpha_\ell G(x_t|m_\ell, \Sigma_t)}, & j \in L_k, \\ 0, & \text{otherwise}; \end{cases}$$

where L_k consists of the first k labels of $\{1, \ldots, k\}$ that correspond the first k largest values of $\alpha_j G(x|m_j, \Sigma_j)$.

It leads to the RPCL learning again when $k = 2$ (Xu, 2001b and 2002).

3.2. Best modeling via a parametric set

Instead of representing multiple data sets with each Gaussian density for a data set, a multi-sets-mixture is more suitable for the fields of computer vision and image recognition, where we often encounter the tasks of detecting objects in typical shapes such as lines, circles, and ellipses, as well as pre-specified shapes. In these cases, we need to model multiple data sets with samples of each data set coming from an object of a given shape. Except for the simplest cases such as points and lines, a Gaussian density is not able to represent such a data set. The multi-sets-mixture or called multi-sets modeling is proposed for modeling these objects (Xu, 1994, 1995b).

Samples from each object consists of one deterministic part plus random noise. The deterministic part is described by a finite or continuous set $S(\theta)$ of real points in R^d, subject to a parametric set θ of a finite number of unknown parameters. Each $S(\theta)$ represents a shape such as a line, a curve, and an ellipsis, as well as a pre-specified shape.

Subject to such a set $S(\theta)$, a sample x is represented by

$$\hat{x} = \arg\min_{y \in S(\theta)} \varepsilon(x, y),$$
$$\varepsilon(x, y) = \mathscr{C}(e(x, y)), e(x, y) = x - y, \tag{29}$$

where $\hat{x}$ is called the best reconstruction of x by $S(\theta)$, and $e(x, \theta) = e(x, \hat{x})$ is called the reconstruction error of $S(\theta)$ per sample x. Moreover, $\varepsilon(x, y)$ is a given measure for the discrepancy $e(x, y)$ such that $\varepsilon(x, y) \geq 0$ and $\varepsilon(x, y) = 0$ if and only if $e(x, y) = 0$ or $x = y$. The best modeling of $S(\theta)$ on a given set of samples is made by determining θ such that

$$\min_{\theta} \sum_{t=1}^{N} \min_{y \in S(\theta)} \varepsilon(x_t, y), \tag{30}$$

The most widely used $\varepsilon(x, y)$ is

$$\varepsilon(x, y) = e(x, y)^{\mathrm{T}} \Sigma^{-1} e(x, y), \tag{31}$$

which is called the Mahalanobis distance with Σ being positively defined. It returns to the square distance between x, y when $\Sigma = I$. In this case, we call $\hat{x}$ the least square reconstruction of x by $S(\theta)$, with

$$\varepsilon_2(x, \theta) = \|e(x, \theta)\|^2 = \min_{y \in S(\theta)} \|e(x, y)\|^2. \tag{32}$$

which is generally a minimizing procedure. However, $e(x, \theta)$ gets an explicit expression in the following special cases:

(a) a point $S(\theta) = \{a\}$: $e(x, \theta) = x - a$; (33)

(b) a line $S(\theta) = \{x : x - a \text{ parallels } w - a\}$,

$$e(x, \theta) = [I - (w - a)(w - a)^{\mathrm{T}}](x - a);$$

(c) a plane $S(\theta) = \{x : (x - a)^{\mathrm{t}}(w - a) = 0\}$,

$$e(x, \theta) = \frac{(w - a)^{\mathrm{T}}(w - a)}{\|w - a\|};$$

(d) a subspace $S(\theta)$ spanned by W at the origin a,

$$e(x, \theta) = (I - W(W^{\mathrm{t}}W)^{-1}W^{\mathrm{t}})(x - a);$$

(e) a circle $S(\theta) = \{x : \|x - a\|^2 = c^2\}$,

$$e(x, \theta) = \|x - a\| - c;$$

(f) an ellipse

$$S(\theta) = \{x = [u, v]^{\mathrm{T}} : \frac{(u - u_0)^2}{a^2} + \frac{(v - v_0)^2}{b^2} = 1\},$$

$e(x, \theta) = x - \hat{x}$, with $\hat{\mathrm{x}}$ as in Eq.(29).

$S(\theta)$ in the cases (b) and (c) consists of a line or a hyperplane, respectively, passing through a point a, and $S(\theta)$ in the case (d) consists of a *linear manifold*—a shifted subspace that locates at a point a. $S(\theta)$ in the case (e) consists of a *sphere* of radius c that locates at a. The corresponding error $\varepsilon_2(x, \theta)$ is actually the shortest distance of the point x to the line, hyperplane, linear manifold, sphere, and ellipse, respectively.

For the cases (a), (b), (c) and (d), the implementation of Eq. (30) can be made with an analytical solution, with a being the mean vector, w being the direction of either principal or minor component of the sample set, and W spans a principal subspace. For the case (e) and case (f), the implementation of Eq. (30) can be implemented via an iterative procedure, e.g. gradient descent. For all the cases, the implementation of Eq. (30) can also be made adaptively with the parameter θ updated per sample x_t via the descent direction of the gradient $\nabla_\theta \varepsilon_2(x_t, \theta)$.

More generally, given a set of samples $\mathscr{Y} = \{y_r\}_{r=1}^N$ that represents a contour of a specific shape, we have

$$S(\theta) = \{\lambda R(\phi)(y_r + a) : \ \forall y_r \in \mathscr{Y}\} \tag{34}$$

for a shape resulted from a displacement a, a rotation of an angle ϕ and a scaling by λ, where $R(\phi)$ is a rotation matrix and $\theta = \{a, \phi, \lambda\}$. Correspondingly, fitting the shape by Eq. (30) becomes

$$\min_{\theta} \sum_{t=1}^{N} \min_{y_r \in \mathscr{Y}} \|x_t - \lambda R(\phi)(y_r + a)\|^2. \tag{35}$$

3.3. Multi-sets-mixture and adaptive learning

When samples come from multiple objects, a number of $S(\theta_\ell)$, $\ell = 1, \cdots, k$ are needed. As shown in (Xu, 1994, 1995b), the multiple counterparts by the cases of (a), (b), (c) and (d) in Eq. (33) actually perform the mean square error k-means clustering, local principal component analysis (PCA), minor principal component analysis (MCA), local

principal subspace analysis (PSA), and as well as its complementary local MSA.

Directly considering x, a multi-sets will correspond a mixture of non-Gaussian densities that is not easy to implement (Xu, 1995b, 1996a). However, considering the reconstruction error $e(x, \theta_\ell)$ from a Gaussian, we have the following Gaussian mixture

$$p(e|\theta) = \sum_{\ell=1}^{k} \alpha_\ell G(e|0, \Sigma_\ell), \tag{36}$$

which is a further extension of the mixture of exponential densities in (Xu, 1996a).

For each sample x_t, we get $e_{\ell,t} = e(x_t, \theta_\ell)$ for each $S(\theta_l)$ with the probability $p_{\ell,t} = p(\ell|x_t)$, and make learning with data smoothing regularization by maximizing

$$\begin{aligned} L_D(\theta, h) &= \sum_{\ell=1}^{k}\left[\frac{1}{N}\sum_{t=1}^{N} p_{\ell,t} L_{\ell,t}(h) + Z_\ell(h)\right], \\ L_{\ell,t}(h) &= \int G(e|e_{t,\ell}, h^2 I)\ln[\alpha_\ell G(e|0, \Sigma_\ell)]de, \end{aligned} \tag{37}$$

where $Z_\ell(h)$ is same as in Eq. (8) with $e_{t,\ell}$ in place of u_t.

With h^2 fixed, learning can be modeled via the smoothed EM algorithm by Eq. (23) with $G_t(\theta_\ell) = G(e_{\ell,t}|0, \Sigma_\ell)$ and $e(x_t, m_\ell)$ replaced by $e(x_t, \theta_\ell)$, as well as

$$m_\ell = \frac{1}{N\alpha_\ell}\sum_{t=1}^{N} p_{\ell,t} x_t$$

replaced by

$$\theta_\ell = \arg\max_{\hat{\theta}_\ell} \frac{1}{N}\sum_{t=1}^{N} p_{\ell,t} e(x_t, \hat{\theta}_\ell)^{\mathrm{T}} \sum_{\ell}^{-1} e(x_t, \hat{\theta}_\ell). \tag{38}$$

which is usually solvable for those cases in Eq. (33) (Xu, 1994, 1995b).

Then, h^2 is updated by Eq. (10) or Eq. (11) or Eq. (13), with π_q still given by Eq. (24) and $\mathrm{d}L_D(\theta, h)/\mathrm{d}h$ by Eq. (12) but getting

$$p_{t,\tau} = \begin{cases} \dfrac{1}{N}\dfrac{\sum_{j=1}^{k} p_{j,t}\exp\left(\dfrac{-\|e_{j,t} - e_{jt}\|^2}{h^2}\right)}{\sum_{t=1}^{N}\sum_{j=1}^{k} p_{j,t}\exp\left(\dfrac{-\|e_{j,t} - e_{jt}\|^2}{h^2}\right)}, & \text{(a)}, \\ \dfrac{\sum_{\ell=1}^{k}\sum_{j=1}^{k} p_{j,\tau} p_{\ell,t}\exp\left(\dfrac{-\|e_{l,t} - e_{j\tau}\|^2}{h^2}\right)}{\sum_{\tau=1}^{N}\sum_{t=1}^{N}\sum_{\ell=1}^{k}\sum_{j=1}^{k} p_{j,\tau} p_{\ell,t}\exp\left(\dfrac{-\|e_{l,t} - e_{j\tau}\|^2}{h^2}\right)}, & \text{(b)}, \end{cases}$$

Moreover, after parameter learning we can select k by the minimum of $J(k)$ via Eq. (25). Also, similar to the case of Eq. (22), we can replace the E step by Eq. (26) such that an appropriate number k is determined automatically during learning. With $G_t(\theta_\ell) = G(e_{\ell,t}|0, \Sigma_\ell)$ and $e_{\ell,t} = e(x_t, \theta_\ell)$, we get various types of adaptive algorithms as in Eq. (28). Particularly, when $\Sigma_\ell = \sigma_\ell^2 I$, the step (c) in Eq. (28) is simplified as follows

$$\begin{aligned} \sigma_\ell^{\mathrm{new}} &= \sigma_\ell^{\mathrm{old}} - \frac{\eta p_{\ell,t}}{\sigma_\ell^{\mathrm{new}}}\left[1 - \frac{\|e(x, \theta_\ell)\|^2}{\sigma_\ell^{2\mathrm{old}}}\right], \\ \sigma_\ell^{2\mathrm{new}} &= \sigma_\ell^{\mathrm{new}} * \sigma_\ell^{\mathrm{new}}, \\ \theta_\ell^{\mathrm{new}} &= \theta_\ell^{\mathrm{old}} + \eta\nabla_{\theta_\ell}\|e(x, \theta_\ell)\|^2. \end{aligned} \tag{39}$$

A preliminary case of Eqs. (28) and (39) at $\alpha_\ell = 1/k$ and $\Sigma_\ell = \beta I$ for a given $\beta > 0$ was proposed in (Xu, 1998c) for RPCL learning on a multi-sets.

4. Guess-test decision, competiton-penalty adaptation, and global evidence accumulation

The tasks of learning or modeling or problem solving in general can be understood mathematically as a mapping from the observation space $\mathcal{X}$ to a set Θ of hypotheses. A set $\{x_t\}$ comes from $\mathcal{X}$ with each $x \in \mathcal{X}$ called a sample or an evidence. The set Θ is either a finite or infinite set that represents a family of models with a given form of mathematical function and a parameter vector θ such that each fixed value of θ is an element or called a point of Θ, i.e. $\theta \in \Theta$ is a hypothesis that represents a specific mathematical model. A mapping from $\{x_t\}$ to one or several points in Θ indicates that one or several hypotheses are drawn as conclusions from the evidences in $\{x_t\}$. Generally, there are three fundamental strategies for these tasks:

(a) *Competition-penalty adaptation*. As discussed in this paper, we have $\theta_\ell \in \Theta$, $\ell = 1, \ldots, k$ that compete to adapt $\{x_t\}$ in $\mathcal{X}$ via making each θ_ℓ of the k variables move in Θ. All the movements are motivated to minimize a given cost function $C(\{x_t\}, \{\theta_\ell\})$ in a way that the trace of each moving variable is a continuous trace of improvements from a current hypothesis locally to a nearby hypothesis. Finally, the trace is trapped at a hypothesis that any moving to its nearby points has no improvement. Then, this hypothesis is taken as a conclusion. The advantage of this process is easy in implementation with only a small computing cost and memory. Also, it is adapted once an new evidence comes and we always have k hypotheses as current interpretations available. A main disadvantage is thus that the movements are made locally to neighbors only and thus likely trapped at conclusions of local optimal instead of being the best. Moreover, a strong competition among the k variables of θ_k will make the situation even worse. Usually one or more type of penalty is companied with this competition to reduce its negative effect.

(b) *Global evidence accumulation based selection*. As used by the well known Hough Transform (Hough, 1962), each point in Θ is considered as a candidate hypothesis. Each evidence x_t casts one vote to a subset of points in Θ as possible hypotheses. After voted by all evidences in $\{x_t\}$,

those points that receive the number of votes large enough are taken as conclusions. The advantage of the approach is the conclusion is made via a global voting based on all evidences in $\{x_t\}$. The disadvantages is that the cost is very expensive in casting and storing all the votes, usually increasing exponentially with the dimension of θ.

(c) *Guess-test based decision*. As used by the RANSAC approach (Fischler and Bolles, 1981), a point θ of Θ is guessed from a few evidences in $\{x_t\}$, then a given testing criterion is used to check how many samples in $\{x_t\}$ are fitted by the model with this θ and how well this fitting. The guess is either taken as a conclusion if the test is passed or discarded if failed. Then, the next guess-test circle repeats. This approach has little memory cost. However, a simple algorithm that bases on only a few evidences is easy to give a wrong guess and too many wrong guess will cost a lot of computations. In contrast a complicated algorithm needs many computing cost but may not be able to considerably increase the accuracy of guessing. Similarly, a simple testing criterion will create many wrong solutions while a complicated criterion will waste a lot of computing times. Moreover, a testing made on a hypothesis may only give a local solution, instead of an optimal solution. Hence, many open issues remain, especially on making a testing on multiple hypotheses.

It can be observed that the above processes of learning, voting a subset of candidates, and guessing share a common point of providing certain hypotheses as candidates. A difference is that these hypotheses are simply taken as conclusions for (a) but will be further evaluated for (b) and (c). Also, though (b) and (c) share a common point that both evaluate candidate hypotheses before taking any conclusion, a difference is whether conclusions are based either on comparison after enough evidence accumulated for (b) or on a testing immediately following a guessing for (c).

Three strategies can all be regarded as the particular cases of **a general problem solving paradigm** that consists of

(i) Drawing candidate hypotheses from samples $\{x_t\}$,
(ii) Accumulating votes on the candidate hypotheses,
(iii) Selecting most likely hypotheses via comparison,
(iv) Testing the most likely hypotheses that become conclusions or rejected;
(v) Refining the conclusions (e.g. via local adaptation).

For a specific problem solving strategy, some of the above five ingredients may disappear, and the strength of each ingredient may be different. For examples, the above (a) consists of only the ingredient (i), made via a sophisticated learning process. The above (b) consists of the first three ingredients with (i) being enumerating simply each sample in $\{x_t\}$. The above (c) consists of the ingredient (iv) and the ingredient (i) that is simply random sampling.

By putting the focuses and strengths differently on the five ingredients, we may combine the advantages of the above (a) (b) and (c) and reduce the effect of their disadvantages. For an example, the Random Hough Transform (Xu & Oja, 1993) improves the disadvantages of the Hough Transform (Hough, 1962) via modifying the ingredient (i) to reduce the burden of the ingredient (ii) as well as via adding the ingredient (iv) to improve accuracy. For another example, we can implement the above (a) of learning that starts at different initializations to take the role of the ingredient (i), followed by all or a part of all the rest ingredients. Particularly, with learning on a mixture of multi-sets by Eqs. (34) and (35) as the ingredient (i), we can get an improved version of the generalized Hough Transform (Ballard, 1981).

5. Conclusions

The data smoothing based regularization not only provides an easy implementing solution to the difficulty of determining an appropriate hyper-parameter in Tikhonov like regularization for parameter learning, but also takes an important role in BYY harmony learning both on penalizing the WTA effect of the least complexity nature and on improving the performance of model selection criteria in the cases of a small size of samples. The roles are detailed via Gaussian-mixture with the smoothed EM algorithm, the smoothed BYY model selection criterion, adaptive algorithm as well as its related RPCL learning. Moreover, the studies are further extended to a reconstruction error based Gaussian mixture for multi-sets learning with data smoothing based regularization, which are suitable to tasks of modeling and recognizing multiple objects of typical geometrical shapes. Finally, insights are provided on three problem solving strategies, namely the competition-penalty adaptation based learning, the global evidence accumulation based selection, and the guess-test based decision, under the general problem solving paradigm.

Acknowledgements

The work described in this paper was fully supported by a grant from the Research Grant Council of the Hong Kong SAR (project No: CUHK4336/02E).

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*, 714–723.

Ballard, D. H. (1981). Generalizing the Hough transform to detecting arbitary shapes. *Pattern Recognition*, *13*(2), 111–122.

Bishop, C. M. (1995). Training with noise is equivalent to Tikhonov regularization. *Neural Computation*, *7*, 108–116.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum-likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society*, *B39*, 1–38.

Devroye, L., Györfi, L., & Lugosi, G. (1996). *A probability theory of pattern recognition*. Berlin: Springer.

Fischler, M. A., & Bolles, R. C. (1981). Random sample consensus: A paradigm for modeling fitting with applications to image analysis and automated cartography. *Communications of ACM*, *24*, 381–395.

Girosi, F., Jones, M., & Poggio, T. (1995). Regularization theory and neural architectures. *Neural Computation*, *7*, 219–269.

Hough, P. V. C (1962). *Method and means for recognizing complex patterns*. US Patent No. 3069654, December 18

Hu, X. L., & Xu, L. (2003). A comparative study of several cluster number selection criteria. *Proceedings of IDEAL03, March 21–23, Hong Kong*, in press.

Redner, R. A., & Walker, H. F. (1984). Mixture densities, maximum likelihood, and the EM algorithm. *SIAM Review*, *26*, 195–239.

Rissanen, J. (1999). Hypothesis selection and testing by the MDL principle. *Computer Journal*, *42*(4), 260–269.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, *6*, 461–464.

Tikhonov, A. N., & Arsenin, V. Y. (1977). *Solutions of ill-posed problems*. Baltimore, MD: Winston.

Xu, L. (1994). Multisets modeling learning: An unified theory for supervised and unsupervised learning. *Proceedings of IEEE ICNN94, June 26–July 2, 1994, Orlando, FL*, *I*, 315–320.

Xu, L. (1995a). Bayesian–Kullback coupled Ying–Yang machines: Unified learnings and new results on vector quantization. *Proceedings of ICONIP95, Oct 30–Nov 3, 1995, Beijing, China*, 977–988.

Xu, L. (1995b). A unified learning framework: Multisets modeling learning. *Proceedings of the 1995 World Congress on Neural Networks, July 17–21, 1995, Washington, DC*, *I*, 35–42.

Xu, L. (1996a). Bayesian–Kullback Ying–Yang learning scheme: Reviews and new results. *Proceedings of ICONIP96, Sept 24–27, 1996, Hong Kong*, *1*, 59–67.

Xu, L. (1996b). A unified learning scheme: Bayesian–Kullback Ying–Yang machine. *Advances in Neural Information Processing Systems*, *8*, 444–450.

Xu, L. (1997a). Bayesian Ying–Yang system and theory as a unified statistical learning approach: (I) Unsupervised and semi-unsupervised learning. In S. Amari, & N. Kassabov (Eds.), *Brain-like computing and intelligent information systems* (pp. 241–274). Berlin: Springer.

Xu, L. (1997b). Bayesian Ying–Yang system and theory as a unified statistical learning approach: (II) From unsupervised learning to supervised learning and temporal modeling. In K. W. Wong, I. King, & D. Y. Leung, (Eds.), *Theoretical aspects of neural computation: A multidisciplinary perspective* (pp. 25–42). Berlin: Springer.

Xu, L. (1997c). Bayesian Ying–Yang system and theory as a unified statistical learning approach: (II): Models and algorithms for dependence reduction, data dimension reduction, ICA and supervised learning. In K. W. Wong, I. King, & D. Y. Leung, (Eds.), *Theoretical aspects of neural computation: A multidisciplinary perspective*. Berlin: Springer.

Xu, L. (1997d). Bayesian Ying–Yang machine, clustering and number of clusters. *Pattern Recognition Letters*, *18*(11), 1167–1178.

Xu, L. (1998a). Bayesian Ying–Yang system and theory as a unified statistical learning approach (VII): Data smoothing. *Proceedings of ICONIP'98, Oct 21–23, 1998, Kitakyushu, Japan*, *1*, 243–248.

Xu, L. (1998b). BKYY three layer net learning, EM-like al-gorithm, and selection criterion for hidden unit number. *Proceedings of ICONIP'98, Oct 21–23, 1998, Kitakyushu, Japan*, *2*, 631–634.

Xu, L. (1998c). Rival penalized competitive learning, finite mixture, and multisets clustering. *Proceedings of IJCNN98, May 5–9, 1998, Anchorage, Alaska*, *II*, 2525–2530.

Xu, L. (1999). BYY data smoothing based learning on a small size of samples. *Proceedings of IJCNN'99, Washington, DC, USA, July 10–16*, *1*(6), 546–551.

Xu, L. (2000a). Temporal BYY learning for state space approach, hidden Markov model and blind source separation. *IEEE Transactions on Signal Processing*, *48*, 2132–2144.

Xu, L. (2000b). BYY learning system and theory for parameter estimation, data smoothing based regularization and model selection. *Neural, Parallel and Scientific Computations*, *8*, 55–82.

Xu, L. (2001a). BYY harmony learning, independent state space and generalized APT financial analyses. *IEEE Transactions on Neural Networks*, *12*(4), 822–849.

Xu, L. (2001b). Best harmony, unified RPCL and automated model selection for unsupervised and supervised learning on Gaussian mixtures, ME-RBF models and three-layer nets. *International Journal of Neural Systems*, *11*(1), 3–69.

Xu, L. (2002). BYY harmony learning, structural RPCL, and topological self-organizing on mixture models. *Neural Networks*, *15*, 1125–1151.

Xu, L. (2003). BYY learning, regularized implementation, and model selection on modular networks with one hidden layer of binary units. *Neurocomputing*, *51*, 277–301.

Xu, L., Krzyzak, A., & Oja, E. (1993). Rival penalized competitive learning for clustering analysis, RBF net and curve detection. *IEEE Transactions on Neural Networks*, *4*, 636–649.

Xu, L., & Oja, E. (1993). Randomized Hough transform (RHT): Basic mechanisms, algorithms and complexities. *Computer Vision, Graphics, and Image Processing: Image Understanding*, *57*(2), 131–154.