Front. Electr. Electron. Eng. 2012, 7(1): 116–126
DOI 10.1007/s11460-012-0179-x

RESEARCH ARTICLE

Xinbo GAO, Xiumei WANG

Dimensionality reduction with latent variable model

Abstract Over the past few decades, latent variable model (LVM)-based algorithms have attracted considerable attention for the purpose of data dimensionality reduction, which plays an important role in machine learning, pattern recognition, and computer vision. LVM is an effective tool for modeling density of the observed data. It has been used in dimensionality reduction for dealing with the sparse observed samples. In this paper, two LVM-based dimensionality reduction algorithms are presented firstly, i.e., supervised Gaussian process latent variable model and semi-supervised Gaussian process latent variable model. Then, we propose an LVM-based transfer learning model to cope with the case that samples are not independent identically distributed. In the end of each part, experimental results are given to demonstrate the validity of the proposed dimensionality reduction algorithms.

Keywords dimensionality reduction, latent variable model, pairwise constraints, Bregman divergence

1 Introduction

In machine learning and pattern recognition, many interested objects are naturally stored or hidden in very high-dimensional space for better descriptions. However, these descriptions will lead to great troubles for data analysis. On the one hand, the high-dimensional storages will bring great troubles for the data visualization; on the other hand, the high dimensions would lead to "curse of dimensionality" in data analysis and processing. The key technique to avoid above two problems is dimensionality reduction. In mathematics, dimensionality reduction is the process of reducing the number of random variables under specific consideration, that is, the aim of dimensionality reduction is seeking a set of optimal bases for expressing the data. In the past decades, dimensionality reduction techniques have been extensively studied for various applications, e.g., data visualization [1], pattern classification [2–4], and multimedia information retrieval [5,6].

According to the observability of variables, the dimensionality reduction algorithms can be divided into two general categories. One is observed framework, in which the parameters of the dimensionality reduction model are directly computed with the observed samples. In this category, the low-dimensional variables do not appear in computing parameters. The other category is latent variable framework, in which some assumptions about low-dimensional variables will be given, and the mapping function is established from latent variables to the observed samples. The parameters of the dimensionality reduction model are determined according to the density of the observed samples.

In the observed framework, the representative dimensionality reduction model is principal component analysis (PCA), which is an orthogonal basis transformation; PCA is a linear and unsupervised model [7,8]. In order to deal with the nonlinear case, many nonlinear dimensionality reduction methods have been investigated. Examples include kernel PCA [9], multidimensional scaling (MDS) [10], locally linear embedding (LLE) [11,12], Laplacian eigenmap (LE) [13], Isomap [14,15], locality preserving projections (LPP) [16,17], and neighborhood preserving embedding (NPE) [18]. This kind of nonlinear dimensionality reduction methods, such as MDS, LLE, and LPP, also belongs to "manifold learning". Manifold learning is a branch of nonlinear dimensionality reduction, which assumes that the data distribution lies on a nonlinear manifold. To use the supervised information, for example, labels of the samples, some supervised dimensionality reduction methods are proposed one by one, such as, the linear discriminant analysis [19], kernel Fisher discriminant analysis (kernel FDA) [20,21], marginal Fisher analysis (MFA) [22], and local Fisher discriminant analysis (LFDA) [23]. These methods can find the nonlinear compact representation of the high-dimensional data.

Received October 9, 2011; accepted November 22, 2011

Xinbo GAO (✉), Xiumei WANG
School of Electronic Engineering, Xidian University, Xi'an 710071, China
E-mail: xbgao@mail.xidian.edu.cn

In the latent variable model (LVM)-based dimensionality reduction framework, it first assumes that there exist latent variables corresponding to high-dimensional samples in a low-dimensional space, and then some assumptions are made for the latent variables. The mapping function will be offered to build the relationship between latent variables and observed samples through the mapping function. Finally, the probability distribution function of the observed samples can be represented with latent variables and their parameters. The latent variables and their parameters would be obtained through maximizing the probability distribution function. LVM-based dimensionality reduction methods include probabilistic principal component analysis (probabilistic PCA) [24], factor analysis (FA) [25], generative topographic mapping (GTM) [26], and Gaussian process latent variable model (GP-LVM) [27,28]. This kind of methods can exactly model density of the observed samples, even when samples are really sparse or the samples are particularly noisy. In this paper, we focus our attention on this kind of dimensionality reduction methods. The detailed introduction about LVM-based dimensionality reduction methods will be given in Section 2, and Fig. 1 gives an overview of some representative dimensionality reduction methods.

2 Background

In this section, we briefly introduce two dimensionality reduction models, including latent variable model and Gaussian process latent variable model. More formally, let $Y = [y_1, y_2, \ldots, y_N]^{\mathrm{T}}$ be the matrix denoting N observed examples, i.e., the high-dimensional data set to be processed. Each object y_i is described by a D-dimensional feature vector with $y_i \in R^D$. We use $X = [x_1, x_2, \ldots, x_N]^{\mathrm{T}}$ to denote the low-dimensional set with x_i representing positions in latent space of the corresponding high-dimensional point, $x_i \in R^q, q \ll D$.

2.1 Latent variable model

Latent variable model (LVM) is an effective dimensionality reduction approach through modeling the probabilistic distribution of observed samples with involving additional latent variables. According to the distribution of latent variables, the LVM can be divided into two groups: discrete or continuous versions [29]. The representative discrete LVMs are hidden Markov model (HMM) [30] and latent Dirichlet allocation (LDA) model [31,32]. HMM is one of the simplest dynamic Bayesian networks and has been extensively used in speech recognition [33,34] and biometrics feature analysis [35,36]. LDA is a discrete dimensionality reduction model proposed in 2003. It is mainly used in document topic analysis [37,38] and image retrieval [39–41]. The representative continuous LVMs include probabilistic PCA and GP-LVM. In the following, GP-LVM is used as an example for explaining the process of dimensionality reduction with LVM.

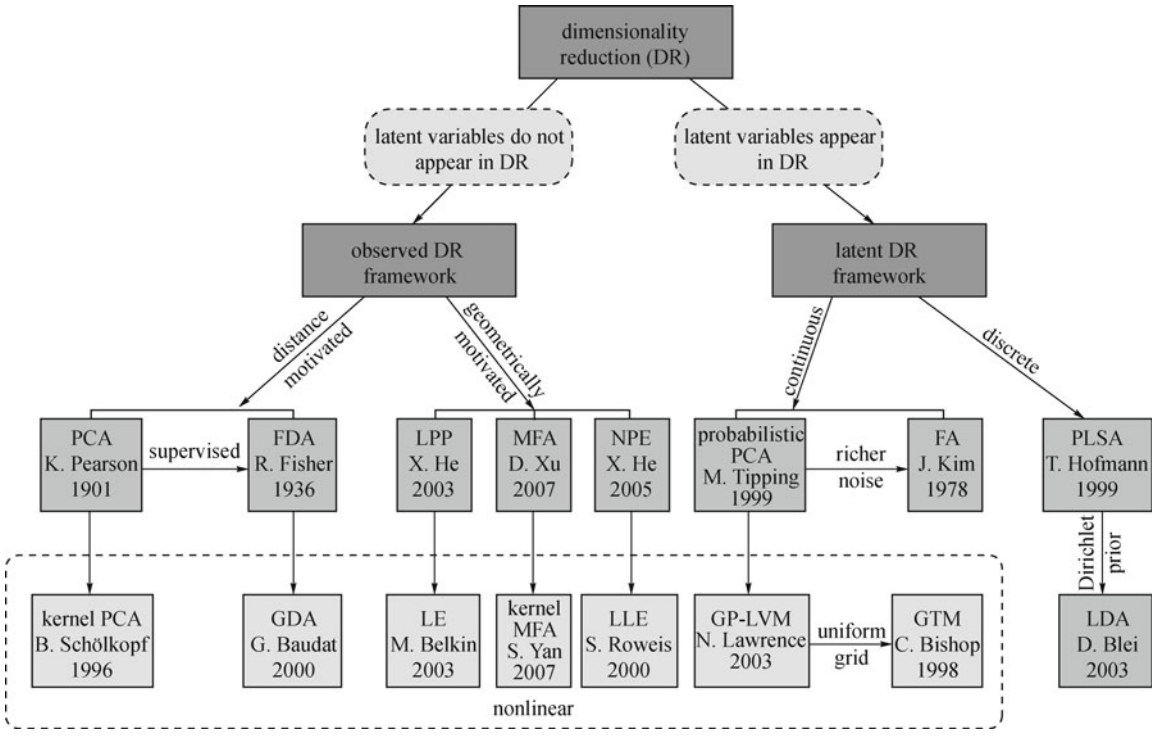

Fig. 1 An overview of some representative dimensionality reduction methods

2.2 Typical GP-LVM

GP-LVM is a nonlinear probabilistic dimensionality reduction algorithm proposed by Lawrence [27]. It can recover complex manifolds through modeling the joint distribution of the observed data. In recent years, it has been widely used in human motion tracking [42,43], object categorization [44,45], and 3D shape estimation [46]. The main reason for receiving considerable attention is that GP-LVM can catch the accurate representation in latent space even if the observed data is very sparse.

The traditional LVMs, for example, probabilistic PCA, marginalize the latent variables and optimize the parameters via maximum likelihood. However, the GP-LVM marginalizes the hyper-parameters and optimizes the latent variables through maximizing likelihood of the observed samples. The positions of the data in the latent space can be obtained through integrating over mapping function f and maximizing likelihood function of the observed data set.

The mapping function $f : X \to Y$ is a Gaussian process priori given by

$$f \sim N(0, K), \tag{1}$$

with the covariance between x_i and x_j, and the kernel function value is determined by a Mercer kernel function, for example, the radius basis function (RBF) kernel. The RBF kernel is employed as the nonlinear mapping function, which can be substituted with

$$k(x_i, x_j) = \theta_{\text{rbf}} \exp\left(-\frac{\gamma}{2}(x_i - x_j)^{\text{T}}(x_i - x_j)\right) + \theta_{\text{white}}\delta_{ij}, \tag{2}$$

where $k(x_i, x_j)$ is the element in the ith row and the jth column of the covariance matrix K, and $\delta_{i,j}$ is the Kronecker delta function. $\theta = [\theta_{\text{rbf}}, \gamma, \theta_{\text{white}}]$ is a collector of the kernel parameters. Then, the likelihood for every dimension can be obtained through marginalizing the mapping function.

$$\begin{aligned} p(y_{:,d}|X, \theta) &= \int p(y_{:,d}|X, f_d, \theta)p(f_d)\mathrm{d}f_d \\ &= N(y_{:,d}|0, K). \end{aligned} \tag{3}$$

The likelihood for the whole observed data can be viewed as a product of D number of independent Gaussian processes, and each process is related to a different dimension of the data set. So the observed data likelihood function can be obtained as

$$p(Y|X, \theta) = \frac{1}{(2\pi)^{\frac{DN}{2}}|K|^{\frac{D}{2}}} \exp\left(-\frac{1}{2}\text{tr}(K^{-1}YY^{\text{T}})\right). \tag{4}$$

In the training process, the latent variables and the parameters will be optimized alternately until the algorithm convergence. After the model is determined, the testing points can be directly computed using the parameters.

3 Dimensionality reduction based on latent variable model

In this section, we will present our proposed LVM-based dimensionality reduction methods with the help of supervised and semi-supervised information. Some experimental results are also reported to illustrate their effectiveness.

3.1 LVM-based dimensionality reduction with supervised learning

Supervised learning is the most effective machine learning paradigm of learning a function from training data and its supervised information. With the help of the supervised learning, many models can obtain much more discriminant results. However the typical GP-LVM is based on unsupervised learning, i.e., it does not use any label information for training. As a consequence, it is essential to develop the supervised variations to further improve its performance in dimensionality reduction tasks.

Gao and Wang et al. [47] proposed a supervised Gaussian process latent variable model (supervised GP-LVM) for dimensionality reduction, which is based on the property of the conditional independence in directed graphs, i.e., the label set and the input data are independent given the latent variables in the low-dimensional space. Both the observed data and the class label information are taken into account in supervised GP-LVM. The supervised model establishes mappings from the latent variables to the observed data and the available sample labels, respectively.

We can establish the relationship between pairs of observations associated with corresponding labels (Y, Z) with $Z \in R^{N\times L}$ and latent variables X as

$$\begin{cases} Y = f(X, \theta), \\ Z = g(X, \gamma). \end{cases} \tag{5}$$

The function f with hyper-parameters θ denotes a mapping that transforms latent variables to observations. The function g with hyper-parameters γ transforms latent variables to observation labels. θ and γ are the collection of the hyper-parameters in two projection, i.e., $\theta = [\theta_{\text{rbf}}, \theta_{\text{band}}, \sigma_y^2]$ and $\gamma = [\gamma_{\text{rbf}}, \gamma_{\text{band}}, \sigma_z^2]$.

In supervised GP-LVM, f and g are two Gaussian processes, and latent variables $X \sim N(0, I)$. Then, the posterior of X can be obtained with Bayes' theorem:

$$p(X|Z, Y) = \frac{p(Y, Z|X)p(X)}{p(Y, Z)}, \tag{6}$$

where $p(Y,Z|X)$ is the likelihood of pairs of observations and labels (Y,Z) and $p(Z,Y)$ are the marginal likelihood of all pairs integrated over X. According to the conditional independence, $p(Y,Z|X) = p(Y|X)p(Z|X)$. The log-posterior of $p(X|Z,Y)$ is given by

$$\ln p(X|Z,Y) = \ln p(Y|X) + \ln p(Z|X) + \ln p(X) - \ln p(Y,Z). \quad (7)$$

The last term in the right-hand side of Eq. (7) is log marginal likelihood, so it is irrelevant to the latent variables X and maximizing the log-posterior $p(X|Z,Y)$ is equivalent to maximizing the likelihood function $p(Z,Y|X)$ plus the prior $p(X)$. We denote $\ln p(Y|X)$ as L_Y and $\ln p(Z|X)$ as L_Z.

$$L_Y = \ln p(Y|X) = -\frac{DN}{2}\ln 2\pi - \frac{D}{2}\ln|K| - \frac{1}{2}\mathrm{tr}(K^{-1}YY^{\mathrm{T}}). \quad (8)$$

$$L_Z = \ln p(Z|X) = -\frac{LN}{2}\ln 2\pi - \frac{D}{2}\ln|\Sigma| - \frac{1}{2}\mathrm{tr}(\Sigma^{-1}ZZ^{\mathrm{T}}). \quad (9)$$

The objective function of supervised GP-LVM can be written as

$$\{X,\theta,\gamma\} = \arg\max_{X,\theta,\gamma}\{\ln L_Y + L_Z + \ln p(X)\}. \quad (10)$$

We apply the scaled conjugate gradient (SCG) with regard to latent variables and hyper-parameters for training. The detail optimization algorithm and procedure can be seen in Ref. [47].

In Fig. 2, an intuitionist experimental result is given to show the efficiency of label information. 400 samples are randomly selected for digits "3" and "5" from handwritten digits database USPS, respectively. The dimensionality reduction results for total 800 samples are given in Fig. 2. Figure 2(a) is the result obtained by GP-LVM, and Fig. 2(b) is the result obtained by the supervised GP-LVM. Compared with Fig. 2(a), digits "3" and "5" in Fig. 2(b) can be well separated. The result of the supervised GP-LVM is superior to GP-LVM because it considers the label information in the training stage.

3.2 LVM-based dimensionality reduction with semi-supervised learning

Although supervised learning can achieve discriminative results, labeled data is often limited, and labeling samples is time consuming and requires much human expertise, so it is expensive to obtain supervised information. Semi-supervised learning can take full use of limited supervised information and the abundant unlabeled data to further improve the performance. In Ref. [48], Wang and Gao et al. proposed a semi-supervised LVM with the help of pairwise constraints.

More specifically, two types of pairwise constraints: ***must-link*** constraint and ***cannot-link*** constraint are defined as follows.

- ***Must-link* constraint:** It specifies that two samples should be assigned into one class. The constraints data set can be denoted as

  $$M = \{(y_i, y_j)|y_i \text{ and } y_j \text{ belong to the same class}\}.$$

- ***Cannot-link* constraint:** It specifies that two samples should be assigned into different classes. The constraints data set can be denoted as

  $$C = \{(y_i, y_j)|y_i \text{ and } y_j \text{ belong to different classes}\}.$$

If the pair of samples $(y_i, y_j) \in M$, the latent variables (x_i, x_j) corresponding to (y_i, y_j) will belong to the same class. In the same way, if the pair of samples $(y_i, y_j) \in C$, the latent variables (x_i, x_j) corresponding to (y_i, y_j) will belong to different classes. Then, according to the pairwise constraints relationship among the observed samples and the distances of the observed samples, we can infer the priori information of the latent variables. We

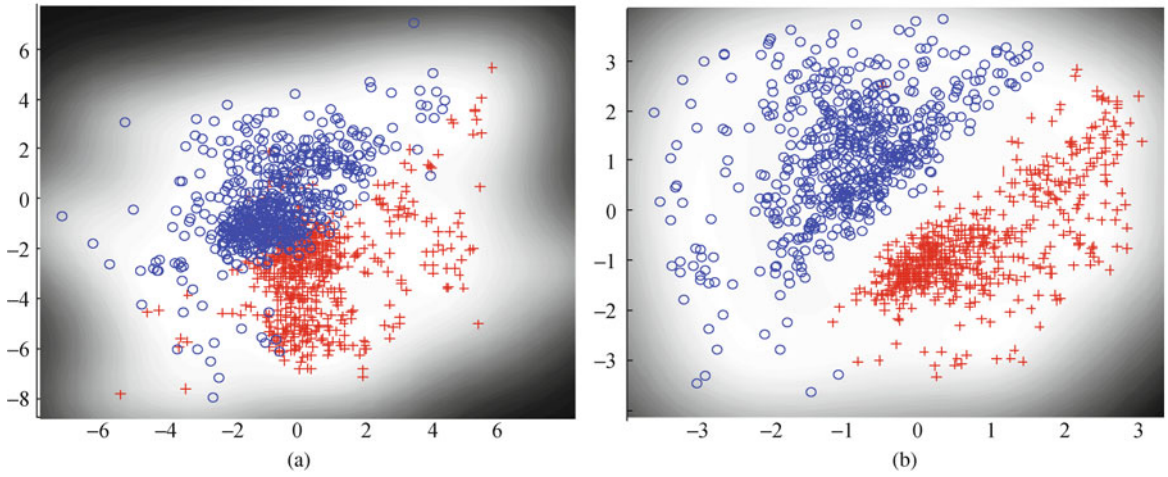

Fig. 2 Dimensionality reduction results for USPS database. (a) Result obtained by GP-LVM; (b) result obtained by supervised GP-LVM

define a weight matrix $W \in R^{N\times N}$ as

$$W_{i,j} = \begin{cases} \dfrac{e^t}{1+e^t}, & (y_i, y_j) \in M, \\ -\dfrac{e^t}{1+e^t}, & (y_i, y_j) \in C, \\ \dfrac{e^t}{N^2(1+e^t)}, & \text{otherwise}, \end{cases} \tag{11}$$

where $t = \|x_i - x_j\|$ represents the Euclidean distance between two latent variables x_i and x_j. N is the number of all samples. In order to emphasize the constraint relationship of the supervised samples, we use N^2 to divide the weights of the unsupervised samples. The value $W_{i,j}$ will be determined by both the distance and the pairwise constraints. If the two samples belong to M, i.e., the same class, the weight value is positive. If they belong to C, the weight value is negative. The weight values will change with the distance among the latent variables as shown in Fig. 3.

As shown in Fig. 3, real line represents the *must-link* relationship in two samples, and dashed line represents *cannot-link* relationship. The values will be positive if the sample pair belongs to same class and negative if the sample pair belongs to different classes. The values are also influenced by the distances of samples.

The priori probability of the latent variables can be defined as

$$P(X) = \frac{1}{Z}\exp\left(-\sum_{i,j=1}^{N} d(x_i, x_j)\right), \tag{12}$$

where $d(x_i, x_j) = W_{i,j} \cdot \|x_i - x_j\|$ and Z is a constant. Here, Z is used to normalize $P(X)$, and make $P(X)$ as a priori probabilistic distribution of the samples. So Eq. (12) can be rewritten as

$$\begin{aligned} P(X|W) &= \frac{1}{Z}\exp\left(-\sum_{i,j=1}^{N} d(x_i, x_j)\right) \\ &= \frac{1}{Z}\exp\left(-\mathrm{tr}(X^{\mathrm{T}}WX)\right) \\ &= \frac{1}{Z}\exp\left(-\mathrm{tr}(WXX^{\mathrm{T}})\right), \end{aligned} \tag{13}$$

where W represents a weight matrix. Just as mentioned above, the weight matrix is defined according to the pairwise constraints and the distances of the samples. Then, the weight matrix W can be obtained by Eq. (13).

Given the constrained priori information of the latent variables, the detailed description of the semi-supervised framework will be given later. The GP-LVM is a latent variable model through defining a joint distribution over the observed variables Y and the latent variables X. The hyper-parameters and the latent variables can be optimized through maximizing the likelihood function as Eq. (4). According to the Bayes' theorem,

$$P(X|Y, \theta, W) = \frac{p(Y, X|\theta)p(X|W)}{p(Y)}. \tag{14}$$

We apply the SCG with regard to latent variables and hyper-parameters for training. The detail optimization process can be seen in Ref. [48]. Equation (13) suggests a general framework for incorporating constraints into the GP-LVM. Particular choices of the pairwise constraints would construct the different weight matrix and produce corresponding algorithms. That is, if the data set can be divided into three parts: *must-link* constrained data, *cannot-link* constrained data, and *unlabeled* data; then, the semi-supervised GP-LVM can be built in three ways as follows:

- **SSGP-LVM-M:** Only the *must-link* constraints is used in the model;
- **SSGP-LVM-CM:** Both the *must-link* and *cannot-link* constraints are used in the model;
- **SSGP-LVM-CMU:** Both the *constrained* sample pairs and the *unlabeled* samples are used for training process.

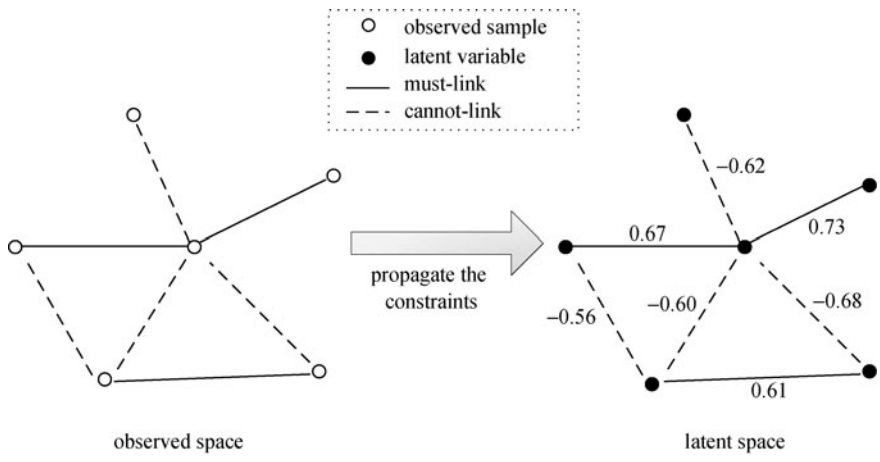

Fig. 3 Transfer scheme of pairwise constraints

To validate the performance of the semi-supervised GP-LVM method, we test the algorithms in different dimensional latent space on three data sets, i.e., ORL face data, Oil data set, and handwritten digits data. The pairwise constraints are randomly selected, and we repeat each experiment 50 times independently. The percent of constraints is given in Table 1.

Table 1 Statistics in percentiles of constraints

data	total number	class	# of constraints
ORL	400	40	4×40
Oil	1000	3	100×3
USPS(3,5)	300×2	2	100×2

As shown in Table 1, the numbers of constrained samples in three data sets are different. Percentage of the constrained samples in each data set is around 30%. The results for three data sets are shown in Fig. 4.

Figure 4 shows the plots for mean error *vs.* number of dimensionalities. The mean errors are decreasing with the increase of the dimensionality of the latent space. That is, these curves have the same decrease tendency. Compared with the traditional GP-LVM, the advantages of SSGP-LVM-M and SSGP-LVM-CM are not obvious as shown in Fig. 4(a). The SSGP-LVM-CMU has much more advantage than other three models. Figures 4(b) and 4(c) show the performance comparison on the Oil data set and USPS data set, respectively. Compared with the GP-LVM, the proposed methods, especially the SSGP-LVM-CM and SSGP-LVM-CMU, significantly outperform the traditional GP-LVM for all the three data sets. As the number of dimensionality grows, the performance of the proposed methods can keep the advantage consistently.

4 Transfer learning with LVM-based dimensionality reduction model

The traditional latent variable models work well under a necessary and strict assumption, that is, both the training and test samples are drawn from the same domain and obey the identical distribution. When they come from different domains, the training and test samples would be not independent identical distribution (*i.i.d.*). Therefore, the performance of the LVMs will be degraded because the parameters of the training model may not suited for the test data set. One attempt to this problem is applying a transfer learning framework to LVMs, which has been verified that it can effectively deal with the different domains problem, i.e., not *i.i.d.*.

For this purpose, Gao and Wang et al. proposed a transfer dimensionality reduction model in Ref. [49]. The model utilizes the Bregman divergence to measure the distance between the training set and test set and then adjusts the model parameters according to the divergence. The details will be given as follows.

4.1 Bregman divergence as distance measurement

Bregman divergence defines a generalization distance to measure the discrepancy between distributions. It has been testified to be effective and efficient in clustering [50] and nearest neighbor retrieval [51].

Definition 1 *Bregman divergence between vectors*: Let $\Phi : \Omega \to R$ be a strictly convex function defined on a closed convex set Ω, and Φ is continuously differentiable. Then, Bregman divergence associated with Φ for vectors $p, q \in \Omega$ is defined as

$$d_\Phi(p, q) = \Phi(p) - \Phi(q) - (\nabla\Phi(q), (p - q)), \qquad (15)$$

where $\nabla\Phi(q)$ denotes the first-order difference of the Φ at point p.

The Bregman divergence can be interpreted as the distance between a function and its first-order Taylor expansion, as shown in Fig. 5.

Besides the vectors, Bregman divergences can also be utilized to measure the distance between matrices, functions, and distributions. The LVMs will be extended to deal with the cross-domain tasks; therefore, the

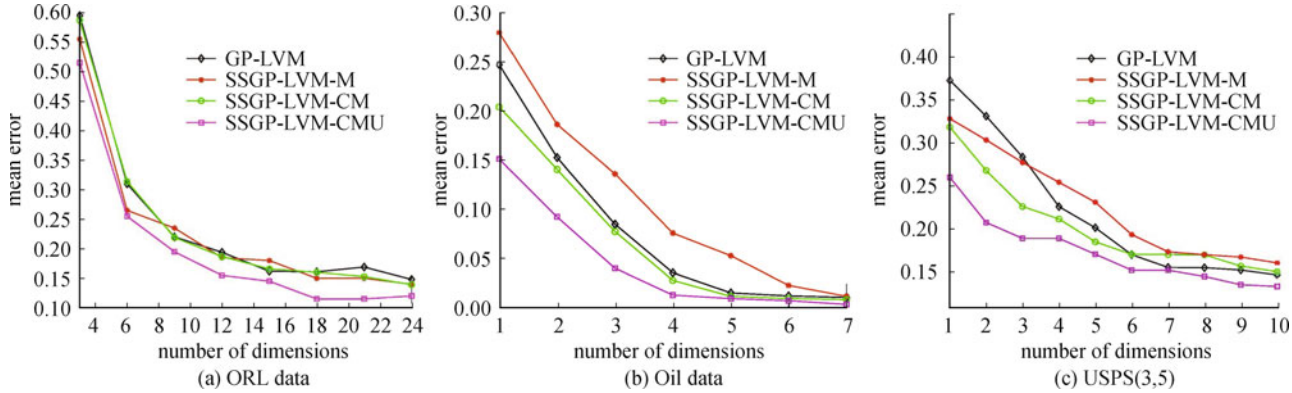

Fig. 4 Comparison of classification error rates between the proposed three methods and the GP-LVM on three data sets with different number of dimensions. (a) ORL data; (b) Oil data; (c) USPS(3,5)

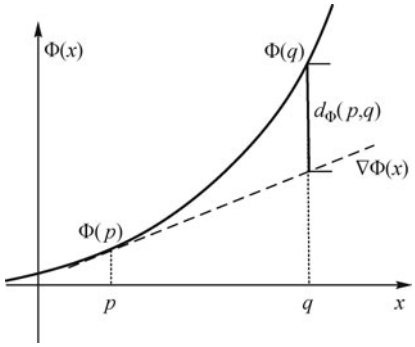

Fig. 5 Bregman divergence between p and q

probability density functions of different domains are also different. Let $p(y)$ and $q(y)$ represent the probability density functions of the training samples Y and test samples Y_t, respectively. The Bregman divergences between the distributions is defined as follows.

Definition 2 *Bregman divergence between distributions*: The Bregman divergence between two given distribution functions $p(y)$ and $q(y)$ under a certain measure μ can be obtained as

$$D_\Phi(p,q) = \int d_\Phi(p(y), q(y))\mathrm{d}\mu(y). \tag{16}$$

As shown in examples, Bregman divergences are generalizations of the squared Euclidean distance that they all share similar properties, such as non-negative and not symmetric. In the next subsection, the Bregman divergence will be utilized to measure the distance between two distributions. If $p(y)$ and $q(y)$ represent the probability density functions, that is, $\int p(y)\mathrm{d}y = 1$ and $\int q(y)\mathrm{d}y = 1$, the divergence will be the *Kullback-Leibler* (KL) divergence. In general, KL divergence is a special case of Bregman divergences.

If Y and Y_t are drawn from different distributions, the samples in two data sets will not be *i.i.d.*. Sometimes, even if the samples were drawn from the same kind of distributions, such as Gaussian distribution, the mean values and covariance matrices of training samples and test samples would not be uniform, e.g., the two Gaussian distributions may have different mean values and covariance matrices.

4.2 Transfer learning with latent variable model

The transfer learning algorithm based on Bregman divergence will be required for measuring the distance between Y and Y_t. Let $Y_t = [y_{t1}, y_{t2}, \ldots, y_{tM}]^{\mathrm{T}}$ denote the test data set and $X_t = [x_{t1}, x_{t2}, \ldots, x_{tM}]^{\mathrm{T}}$ denote the corresponding variables for Y_t in low-dimensional space. Then, a new transfer learning framework for the LVMs can be established by using the regularization as

$$\{X, \theta\} = \arg\min_{X,\theta}\{L(X,\theta) + D(p(Y)\|p(Y_t))\}. \tag{17}$$

As well known, the GP-LVM establishes the Gaussian process mapping from the latent space to each dimension of the observed space, and the likelihood function of the test data set Y_t can be represented as

$$P(Y_t|X_t) = \frac{1}{(2\pi)^{\frac{DM}{2}}|K|^{\frac{D}{2}}}\exp\left(-\frac{1}{2}\mathrm{tr}(K_t^{-1}Y_tY_t^{\mathrm{T}})\right). \tag{18}$$

To estimate the distance between the training and test samples, we just need to measure the distance between the training and test data sets.

$$\begin{aligned} D(p(Y)\|p(Y_t)) &= \sum_{d=1}^{D} KL(Y_{:,d}\|Y_{t(:,d)}) \\ &= \frac{D}{2}\ln|K_tK^{-1}| + \frac{D}{2}\mathrm{tr}\left(K_t^{-1}(K-K_t)\right). \end{aligned} \tag{19}$$

Then, the transfer learning framework for the latent variable model can be formulated as

$$\{X, \theta\} = \arg\min_{X,\theta}\{F(X,\theta)\}. \tag{20}$$

The function $F(X,\theta)$ is equal to

$$\begin{aligned} F(X,\theta) &= L(X,\theta) + D(p(Y)\|p(Y_t)) \\ &= \frac{DN}{2}\ln 2\pi + \frac{D}{2}\ln|K| + \frac{1}{2}\mathrm{tr}(K^{-1}YY^{\mathrm{T}}) \\ &\quad + \frac{D}{2}\ln|K_tK^{-1}| + \frac{D}{2}\mathrm{tr}\left(K_t^{-1}(K-K_t)\right). \end{aligned} \tag{21}$$

The above Eq. (21) can also be rewritten as

$$\begin{aligned} F(X,\theta) &= \frac{DN}{2}\ln 2\pi + \frac{D}{2}\ln(|K|\cdot|K_tK^{-1}|) \\ &\quad + \frac{1}{2}\mathrm{tr}(K^{-1}YY^{\mathrm{T}} + DK_t^{-1}(K-K_t)) \\ &= \frac{DN}{2}\ln 2\pi + \frac{D}{2}\ln(|K_t|) \\ &\quad + \frac{1}{2}\mathrm{tr}(K^{-1}YY^{\mathrm{T}} + DK_t^{-1}K + DE), \end{aligned} \tag{22}$$

where E is an identity matrix.

There is an inevitable instance that the number of the training samples is different with the test samples, i.e., $M \neq N$. In this case, kernel matrices $K \in R^{N\times N}$ and $K_t \in R^{M\times M}$ $(M \neq N)$, and it makes no sense to calculate K_tK^{-1} or $K - K_t$. To handle this issue, we utilize the informative vector machine (IVM) algorithm to extract a subset from the data set [45]. That is, the IVM can represent the data set by a subset I, which is an active set contained r samples, where $r \leqslant \min\{N, M\}$. The objective function can be represented as

$$\begin{aligned} F_I(X,\theta) &= \frac{DN}{2}\ln 2\pi + \frac{D}{2}\ln(|K_{tI}|) \\ &\quad + \frac{1}{2}\mathrm{tr}(K_I^{-1}Y_IY_I^{\mathrm{T}} + DK_{tI}^{-1}K_I + DE_I). \end{aligned} \tag{23}$$

The transfer learning framework is given as

$$\{X, \theta\} = \arg\min_{X,\theta}\{F_I(X, \theta)\}. \quad (24)$$

The Bregman divergence possesses various representations according to the convex function Φ, which modality should be chosen in the transfer learning, will be determined by the distributions of the training and test samples. The framework of the transfer learning process can be divided into two steps.

First, the hyper-parameters and the positions for training samples in the latent space are determined by the traditional GP-LVM through maximizing the likelihood function.

Second, the divergence between the training and test samples are calculated, and then the hyper-parameters can be updated according to the obtained divergence, and the positions of the test samples in the latent space are determined finally.

We empirically investigate the performance of the proposed transfer learning algorithm on two kinds of real-world data sets. One collects three face data sets, i.e., ORL, Yale, and YaleB. The ORL data contains 400 face images, of which 40 individuals are selected as test-bed. For each individual, there are 10 different images taken at different times, varying the lighting and facial expressions. The Yale data contains 165 images of 15 individuals and each has 11 images with different facial expressions or configurations. The YaleB data contains a total 38 (subjects) $\times$ 64 (illumination conditions) samples. The size of each cropped image for three data sets is pixels as shown in Fig. 6.

Fig. 6 Images sampled from Yale, ORL, and YaleB

In order to verify the validity of the proposed transfer learning LVM (TLVM)-based dimensionality reduction algorithm, we compare the performance of the proposed TLVM with the traditional LVM in two sides, e.g., the data reconstruction error and recognition error. In traditional LVM learning, there is no consideration of the relationship between training and test sets. While the transfer learning tasks could build on the cross-domain, and the distance between the training and test data is considered for capturing the similar properties of the two data sets. In the following experiments, the number of active samples is 40 for each face data set. For most dimensionality reduction algorithms, the experimental results usually vary with the number of data dimensions. Therefore, we study the performance of the proposed dimensionality reduction methods varying with different dimensions of the latent space in Fig. 7.

Figure 7 shows the face reconstruction error *vs.* the dimensions of the latent space. With considering the distances between training and test data sets, the transfer learning framework works well under the condition that training and test samples share common properties. TLVM outperforms the traditional LVM on both data sets, especially for the ORL data set. Although the YaleB face data set is an extension of the Yale, the reconstruction error is larger than that on the ORL data set. The reason is that the ORL data set does not contain the variations of lighting conditions, while the YaleB data set contains strong variations of illumination and poses.

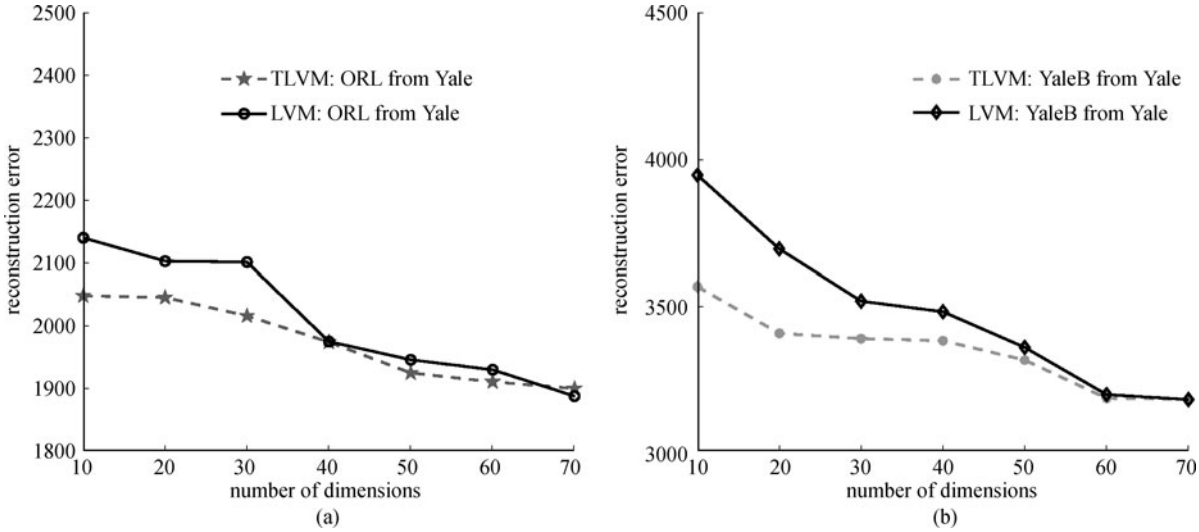

Fig. 7 Comparison of reconstruction error with different dimensions of the latent space on ORL (a) and YaleB (b) from Yale

Figure 8 presents the recognition error rate changing with the different dimensions of the latent space. The nearest neighborhood (NN) classifier is used to test the recognition error rate. The experimental results confirm that the proposed TLVM framework actually outperforms the traditional LVM for cross-domain tasks. For the ORL face data set, the performance of TLVM is improved obviously especially when the dimensions of the latent space are small. For the YaleB face data set, TLVM could also acquire accurate recognition rate. However, as shown in Fig. 8(b), it performs very similar to the traditional LVM. This is because the transfer information from training data set, i.e., Yale face data set, cannot represent so many variations in the YaleB face data set, which contains lighting conditions or postures.

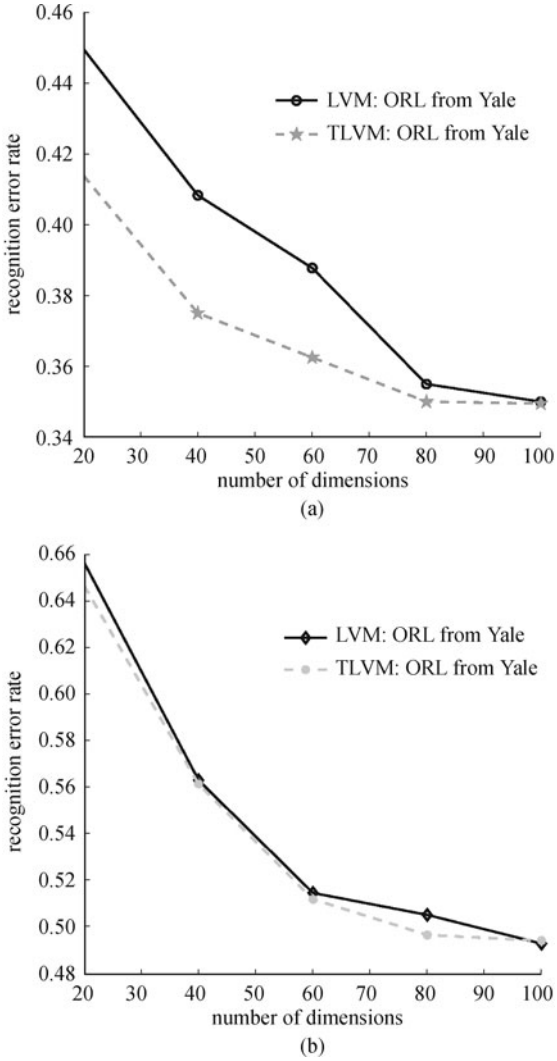

Fig. 8 Comparison of recognition error rate with different dimensions of the latent space on ORL (a) and YaleB (b) from Yale

5 Conclusions

In this paper, we first give an overview on dimensionality reduction algorithms and then present two LVM-based dimensionality reduction algorithms. One is supervised Gaussian process latent variable model, and the other is semi-supervised Gaussian process latent variable model. The former is established on the full labeled data set, and the latter is modeled on the semi-supervised information, i.e., pairwise constraints. The paper also discusses the key techniques in building transfer learning framework based on the LVMs. The key techniques mentioned in this paper can be generalized to other generative models, not only for Gaussian process latent variable model. The reason is that these techniques have generalized properties in the generative models, such as conditional independent, Bayes' theorem, and divergence analysis.

Acknowledgements The authors are thankful for the helpful comments and suggestions from the anonymous reviewers. This research was supported partially by the National Natural Science Foundation of China under Grant Nos. 61125204, 61172146, and 61100158, the Ph.D. Programs Foundation of Ministry of Education of China under Grant 20090203110002, and the Fundamental Research Funds for the Central Universities.

References

1. Koren Y, Carmel L. Robust linear dimensionality reduction. IEEE Transactions on Visualization and Computer Graphics, 2004, 10(4): 459–470
2. Duda R O, Hart P E, Stork D G. Pattern Classification. 2nd ed. New York: Wiley, 2000
3. Tao D, Li X, Wu X, Hu W, Maybank S J. Supervised tensor learning. Knowledge and Information Systems, 2007, 13(1): 1–42
4. Tao D, Li X, Wu X, Maybank S J. General averaged divergences analysis. In: Proceedings of IEEE International Conference on Data Mining. 2007, 302–311
5. He X. Laplacian regularized D-optimal design for active learning and its application to image retrieval. IEEE Transactions on Image Processing, 2010, 19(1): 254–263
6. Tao D, Tang X, Li X, Rui Y. Direct kernel biased discriminant analysis: A new content-based image retrieval relevance feedback algorithm. IEEE Transactions on Multimedia, 2006, 8(4): 716–727
7. Joliffe I. Principal Component Analysis. New York: Springer, 1986
8. Anderson T W. Asymptotic theory for principal component analysis. Annals of Mathematical Statistics, 1963, 34(1): 122–148
9. Schölkopf B, Smola A, Müller K R. Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation, 1998, 10(5): 1299–1319
10. Fiori S. Visualization of Riemannian-manifold-valued elements by multidimensional scaling. Neurocomputing, 2011, 74(6): 983–992
11. Roweis S T, Saul L K. Nonlinear dimensionality reduction by locally linear embedding. Science, 2000, 290(5500): 2323–2326
12. Li X, Lin S, Yan S, Xu D. Discriminant locally linear embedding with high-order tensor data. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2008, 38(2): 342–352

13. Belkin M, Niyogi P. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation, 2003, 15(6): 1373–1396
14. Tenenbaum J B, de Silva V, Langford J C. A global geometric framework for nonlinear dimensionality reduction. Science, 2000, 290(5500): 2319–2323
15. de Silva V, Tenenbaum J B. Global versus local methods in nonlinear dimensionality reduction. Advances in Neural Information Processing Systems, 2003, 15: 705–712
16. He X, Niyogi P. Locality preserving projections. Advances in Neural Information Processing Systems, 2003, 16: 153–160
17. He X, Yan S, Hu Y, Niyogi P, Zhang H-J. Face recognition using Laplacianfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(3): 328–340
18. He X, Cai D, Yan S, Zhang H-J. Neighborhood preserving embedding. In: Proceedings of the 10th International Conference on Computer Vision. 2005, 2: 1208–1213
19. Bar-Hillel A, Hertz T, Shental N, Weinshall D. Learning a Mahalanobis metric from equivalence constraints. Journal of Machine Learning Research, 2005, 6(1): 937–965
20. Mika S, Rätsch G, Weston J, Schölkopf B, Müller K R. Fisher discriminant analysis with kernels. In: Proceedings of IEEE Signal Processing Society Workshop (Neural Networks for Signal Processing IX). 1999, 41–48
21. Baudat G, Anouar F. Generalized discriminant analysis using a kernel approach. Neural Computation, 2000, 12(10): 2385–2404
22. Yan S, Xu D, Zhang B, Zhang H-J, Yang Q, Lin S. Graph embedding and extensions: A general framework for dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(1): 40–51
23. Sugiyama M. Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis. Journal of Machine Learning Research, 2007, 8: 1027–1061
24. Tipping M E, Bishop C M. Probabilistic principal component analysis. Journal of the Royal Statistical Society, Series B: Methodological, 1999, 61(3): 611–622
25. Bartholomew D J. Statistical Factor Analysis and Related Methods. New York: Wiley, 2004
26. Bishop C M, Svensen M, Williams C K I. GTM: The generative topographic mapping. Neural Computation, 1998, 10(1): 215–234
27. Lawrence N D. Gaussian process models for visualization of high dimensional data. Advances in Neural Information Processing Systems, 2004, 16: 329–336
28. Lawrence N D. Probabilistic non-linear principal component analysis with Gaussian process latent variable models. Journal of Machine Learning Research, 2005, 6: 1783–1816
29. Bishop C M. Learning in Graphical Models. Cambridge: MIT Press, 1999
30. Rabiner L R, Juang B H. An introduction to hidden Markov models. IEEE ASSP Magazine, 1986, 3(1): 4–16
31. Blei D M, Ng A Y, Jordan M I. Latent Dirichlet allocation. Journal of Machine Learning Research, 2003, 3: 993–1022
32. Blei D M, Jordan M I. Variational methods for the Dirichlet process. In: Proceedings of the 21st International Conference on Machine Learning. 2004
33. Rabiner L R. A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE, 1989, 77(2): 257–286
34. Baker J K. The DRAGON system — An overview. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1975, 23(1): 24–29
35. Zhong J, Gao X, Tian C. Face sketch synthesis using E-HMM and selective ensemble. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing. 2007, 1: 485–488
36. Gao X, Zhong J, Li J, Tian C. Face sketch synthesis algorithm based on E-HMM and selective ensemble. IEEE Transactions on Circuits and Systems for Video Technology, 2008, 18(4): 487–496
37. Fritz M, Schiele B. Decomposition, discovery and detection of visual categories using topic models. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2008, 1–8
38. Blei D M, Griffiths T L, Jordan M I. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. Journal of the Association for Computing Machinery, 2010, 57(2): 21–30
39. Li F, Perona P. A Bayesian hierarchical model for learning natural scene categories. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2005, 2: 524–531
40. Wang C, Blei D, Li F. Simultaneous image classification and annotation. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2009, 1903–1910
41. Niu Z, Hua G, Gao X, Tian Q. Spatial-DiscLDA for visual recognition. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2011, 1769–1776
42. Wang J M, Fleet D J, Hertzmann A. Gaussian process dynamical models for human motion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(2): 283–298
43. Urtasun R, Fleet D J, Fua P. 3D people tracking with Gaussian process dynamical models. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2006, 1: 238–245
44. Navaratnam R, Fitzgibbon A W, Cipolla R. The joint manifold model for semi-supervised multi-valued regression. In: Proceedings of the 11th IEEE International Conference on Computer Vision. 2007, 1–8
45. Gupta A, Chen T, Chen F, Kimber D, Davis L S. Context and observation driven latent variable model for human pose estimation. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2008, 1–8
46. Salzmann M, Urtasun R, Fua P. Local deformation models for monocular 3D shape recovery. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition. 2008, 1–8
47. Gao X, Wang X, Tao D, Li X. Supervised Gaussian process latent variable model for dimensionality reduction. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 2011, 41(2): 425–434
48. Wang X, Gao X, Yuan Y, Tao D, Li J. Semi-supervised Gaussian process latent variable model with pairwise constraints. Neurocomputing, 2010, 73(10–12): 2186–2195
49. Gao X, Wang X, Li X, Tao D. Transfer latent variable model based on divergence analysis. Pattern Recognition, 2011, 44(10–11): 2358–2366
50. Nock R, Nielsen F. On weighting clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2006, 28(8): 1223–1235
51. Cayton L. Fast nearest neighbor retrieval for Bregman divergences. In: Proceedings of the 25th International Conference on Machine Learning. 2008, 112–119

Xinbo GAO received the B.Eng., M.Eng., and Ph.D. degrees in signal and information processing from Xidian University, China, in 1994, 1997, and 1999, respectively. From 1997 to 1998, he was a research fellow in Department of Computer Science at Shizuoka University, Japan. From 2000 to 2001, he was a postdoctoral research fellow in Department of Information Engineering at the Chinese University of Hong Kong, China. Since 2001, he joined the School of Electronic Engineering at Xidian University. Currently, he is a Professor of Pattern Recognition and Intelligent System, and Director of the Video and Image Processing System (VIPS) Laboratory, Xidian University.

His research interests are machine learning, computer vision, image processing, pattern recognition, and wireless communications. In these areas, he has published five books and around 120 technical articles in refereed journals and proceedings, including *IEEE Transactions on Image Processing*, *IEEE Transactions on Circuits and Systems for Video Technology*, *IEEE Transactions on Systems, Man, and Cybernetics*, *Pattern Recognition*, etc. He is on the editorial boards of journals, including *EURASIP Signal Processing* (Elsevier) and *Neurocomputing* (Elsevier). Dr. Gao served as general chair/co-chair or program committee chair/co-chair or PC member for around 30 major international conferences. Now, he is a Fellow of Institution of Engineering and Technology and a Senior Member of IEEE. He is the Chair of the Multi-valued and Fuzzy Logic Society of the China Computer Federation (CCF) and the Vice Chair of the IEEE Computational Intelligence Society, Xi'an Section.

Xiumei WANG received her Ph.D. degree from Xidian University in 2010. She is currently a lecturer at the School of Electronic Engineering in Xidian University. Her research interests mainly involve nonparametric statistical models and machine learning. In these areas, she has published several scientific articles, including *IEEE TSMC-B*, *Pattern Recognition*, *Neurocomputing* (Elsevier), etc.