Supervised Gaussian Process Latent Variable Model for Dimensionality Reduction

Xinbo Gao, *Senior Member, IEEE*, Xiumei Wang, Dacheng Tao, *Member, IEEE*, and Xuelong Li, *Senior Member, IEEE*

***Abstract*—The Gaussian process latent variable model (GP-LVM) has been identified to be an effective probabilistic approach for dimensionality reduction because it can obtain a low-dimensional manifold of a data set in an unsupervised fashion. Consequently, the GP-LVM is insufficient for supervised learning tasks (e.g., classification and regression) because it ignores the class label information for dimensionality reduction. In this paper, a supervised GP-LVM is developed for supervised learning tasks, and the maximum *a posteriori* algorithm is introduced to estimate positions of all samples in the latent variable space. We present experimental evidences suggesting that the supervised GP-LVM is able to use the class label information effectively, and thus, it outperforms the GP-LVM and the discriminative extension of the GP-LVM consistently. The comparison with some supervised classification methods, such as Gaussian process classification and support vector machines, is also given to illustrate the advantage of the proposed method.**

***Index Terms*—Dimensionality reduction, Gaussian process latent variable model (GP-LVM), generalized discriminant analysis (GDA), probabilistic principal component analysis (probabilistic PCA), supervised learning.**

I. Introduction

THE PURPOSE of dimensionality reduction techniques is to discover the low-dimensional manifold embedded in a high-dimensional space [1]–[5]. So far, many dimensionality reduction approaches are using linear mappings [6]–[9], [11], i.e., dimensionality reduction is realized by performing a linear transformation on samples in the high-dimensional observation space [12], [13]. Examples include factor analysis (FA) and probabilistic principal component analysis (probabilistic PCA); both of which seek the probability distribution and a set of bases for subspace selection [14]–[16]. FA is a popular latent variable model (LVM) which finds a low-dimensional vector of unobserved or latent variables related to a high-dimensional observation vector. Probabilistic PCA is a representative Gaussian LVM which assumes that samples are drawn from a Gaussian distribution. Thus, probabilistic PCA can be deemed as a special FA.

Linear methods are easy to understand and are simple to implement. However, linearity assumption usually leads to a poor performance in many real-world scenarios [17]–[19] because samples are nonlinearly distributed. Therefore, many algorithms have been proposed for nonlinear dimensionality reduction. The kernel PCA is a typical work [20]. It employs the nonlinear kernel technique in the observation space, and then, dimensionality reduction is performed in the kernel space. Although the kernel PCA establishes a nonlinear mapping to transform samples from the observation space to the latent variable space, it is inconvenient to project back the transformed samples from the latent variable space to the observation space. To this end, the Gaussian process LVM (GP-LVM) [21]–[24] is proposed, and it offers a projection to reproduce the transformed samples from the latent variable space to the observation space.

The GP-LVM has been identified to be an effective nonlinear dimensionality reduction algorithm. It is a dual probabilistic interpretation to probabilistic PCA, and it provides a nonlinear mapping to reproduce the transformed samples from the latent variable space (i.e., low-dimensional space) to the observation space by imposing a Gaussian process prior over the mapping function [25]. By modeling the joint probability density of the observed samples, the GP-LVM obtains the low-dimensional manifold with a small number of samples.

All of the aforementioned dimensionality reduction algorithms are unsupervised, i.e., they do not use the label information (e.g., regression values and binary class labels) for training. As a consequence, it is essential to develop the supervised variations to further improve their performance in supervised learning tasks. Recently, the shared LVM has been introduced to GP-LVM [26], [27]. They join latent variables of two observed data sets (e.g., the image features and the joint angles used in [26]). Therefore, they are shared LVMs for two related observed data sets.

In order to utilize the label information of the samples, a discriminative extension of the GP-LVM (discriminative GP-LVM) is proposed in [28]. It used *a priori* information on latent variables based on the generalized discriminant analysis

Manuscript received May 25, 2009; revised February 18, 2010; accepted June 12, 2010. Date of publication August 9, 2010; date of current version March 16, 2011. This work was supported in part by the National Natural Science Foundation of China under Grant 60771068, Grant 60702061, and Grant 60832005; by the Ph.D. Programs Foundation of the Ministry of Education of China under Grant 20090203110002; by the Natural Science Basic Research Plane in Shaanxi, China, under Grant 2009JM8004; by the Open Project Program of the National Laboratory of Pattern Recognition, Shenzhen University, China; and by the National Basic Research Program of China (973 Program) under Grant 2011CB707000. This paper was recommended by Associate Editor Q. Ji.

X. Gao and X. Wang are with the School of Electronic Engineering, Xidian University, Xi'an 710071, China (e-mail: xbgao@mail.xidian.edu.cn; wangxiumei@gmail.com).

D. Tao is with the School of Computer Engineering, Nanyang Technological University, 639798 Singapore (e-mail: dctao@ntu.edu.sg).

X. Li is with the Center for **OPT**ical **IM**agery **A**nalysis and **L**earning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: xuelong_li@opt. ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2010.2057422

(GDA) and used the label information to maximize the between-class separability and to minimize the within-class variability. The discriminative GP-LVM can preserve more discriminative information for classification than the GP-LVM. Since it is a tradeoff between the GP-LVM and the GDA, it would suffer from the disadvantages of both the GP-LVM and the GDA. Given the latent variables, the corresponding observed samples are independent from their label information. Based on the property of the conditional independence in the LVM, a supervised dimensionality reduction method is proposed to improve the performance of the GP-LVM. This conditional independent property is also popular in statistics and data mining. For example, supervised extensions of PCA and probabilistic PCA are developed to generalize the PCA and probabilistic PCA to utilize the label information for classification and to achieve a better performance for many practical classification tasks [29].

In this paper, we propose a supervised GP-LVM 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 [30], [31]. Both the observed data and the class label information are taken into account in the supervised GP-LVM. The supervised model establishes mappings from the latent variables to the observed data and the available sample labels. Then, the hyperparameters for dimensionality reduction and classification can be optimized simultaneously. In the experiments, the advantages of the supervised GP-LVM, compared with the GP-LVM and the discriminative GP-LVM, are testified.

The rest of this paper is organized as follows. In Section II, we summarize some previous related works, such as the LVM and the conditional independence. Section III details how to establish the supervised GP-LVM and its realization algorithm. The experimental results and analysis are presented in Section IV, and Section V concludes this paper.

II. Brief Review of the GP-LVM

In this section, to better understand the proposed supervised GP-LVM, we first brief the GP-LVM. Let $X = [x_1, \ldots, x_N]^T$ be the matrix containing a set of N observations, i.e., the input samples to be processed. Each observation x_i is described by a D-dimensional feature vector $x_i \in R^D$. The dth column of X is denoted as $x_{:,d}$. Let $Y = [y_1, \ldots, y_N]^T$ be the label matrix, wherein $y_i \in R^L$ for the regression problem or $y_i \in \{+1, -1\}$ for binary classification. The lth column of Y is denoted as $y_{:,l}$. Let $Z = [z_1, \ldots, z_N]^T$ be the low-dimensional data set and z_i represent the ith sample in the latent variable space of the corresponding observation x_i in the high-dimensional space. The GP-LVM defines a joint distribution over X and Z.

A. Gaussian Process Prior

Gaussian process is a natural generalization of multivariate Gaussian random variables to infinite index sets. It provides a promising nonparametric Bayesian explanation to metric regression and classification problems [32], [33].

A Gaussian process prior over a function defines a flexible probabilistic distribution. If such a prior is imposed over the mapping function from Z to Y, i.e., $g_{GP} : Z \rightarrow Y$, then we have $g_{GP}(z) \sim GP(m_g(z), k_g(z, z'))$ [25].

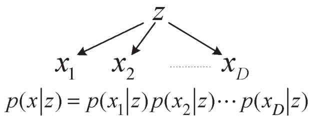


Fig. 1. Conditional independence is illustrated. Given latent variable z, all dimensions of the observations are independent.

Usually, for the sake of simplicity, the mean function $m_g(z)$ is set to zero, and the kernel function between z_i and z_j is defined by a Mercer kernel, e.g., the radius basis function (RBF)

$$k_g(z_i, z_j) = \gamma_{\text{rbf}} \exp\left(-\frac{\gamma_{\text{band}}}{2}(z_i - z_j)^T(z_i - z_j)\right). \quad (1)$$

For more realistic modeling situations, there exists noise in the process of obtaining data set. Therefore, the output can be represented as

$$Y = g_{GP}(Z) + \varepsilon_y \quad (2)$$

where ε_y is an independent Gaussian noise $\varepsilon_y \sim N(0, \sigma_y^2 I)$. The distribution of Y can be obtained, i.e., $Y \sim N(0, \Sigma)$. The entry of the covariance matrix Σ is defined as

$$\Sigma_{i,j} = k_{GP}(z_i, z_j) + \sigma_y^2 \cdot \delta_{i,j}. \quad (3)$$

The relationship between Y and Z can be written as

$$Y = g(Z, \gamma) \quad (4)$$

where $g(Z, \gamma) = g_{GP}(z) + \varepsilon_y$ and γ is the collection of the hyperparameters in the projection, i.e., $\gamma = [\gamma_{\text{rbf}}, \gamma_{\text{band}}, \sigma_y^2]$.

B. LVM

The LVM is a powerful approach in probabilistic modeling, and it involves supplementing a set of observed variables with additional latent variables. The LVM can be divided into two groups, namely, discrete or continuous, according to the distribution of the latent variables [30]. Some dimensionality reduction methods (e.g., PCA, probabilistic PCA, and GP-LVM) are continuous LVM.

C. Conditional Independence

The key point of the LVM is that the dimensions of the observations are conditionally independent with the given latent variable z, as shown in Fig. 1. The conditional probability of the observed sample x can be obtained by

$$p(x|z) = p(x_1|z)p(x_2|z) \cdots p(x_D|z). \quad (5)$$

In this paper, we will use this property to construct the supervised GP-LVM.

D. GP-LVM

The GP-LVM is the dual representation of the probabilistic PCA. The probabilistic PCA imposes the Gaussian distribution over every latent variable as a prior and determines the principal axes of a set of observations by using the maximum-likelihood estimation of the parameters of an LVM [15].

If a Gaussian process prior is imposed over the mapping function f_{GP}, then $f_{GP}(z) \sim GP(m_f(z), k_f(z, z'))$. The RBF is chosen as the kernel function, i.e.,

$$k_f(z, z') = \theta_{\text{rbf}} \exp\left(-\frac{\theta_{\text{band}}}{2}(z - z')^T(z - z')\right). \quad (6)$$

Therefore, each dimension of the output can be represented as

$$X_{:,d} = f(Z, \theta) = f_{GP}(Z) + \varepsilon_x \quad (7)$$

where ε_x is an independent Gaussian noise, i.e., $\varepsilon_x \sim N(0, \sigma_x^2 I)$. θ is the collection of the hyperparameters in mapping function f, i.e., $\theta = [\theta_{\text{rbf}}, \theta_{\text{band}}, \sigma_x^2]$. Therefore, $X_{:,d}$ also obeys normal distribution, i.e.,

$$X_{:,d} \sim N(0, K(Z, \theta)) \quad (8)$$

where $K_{i,j} = k_f(z_i, z_j) + \sigma_x^2 \cdot \delta_{i,j}$. The marginal likelihood for every dimension is given by

$$\begin{aligned} p(X_{:,d}|Z, \theta) &= \int p(X_{:,d}|Z, f)p(f)df \\ &= N(X_{:,d}|0, K(Z, \theta)). \end{aligned} \quad (9)$$

Therefore, the likelihood of a set of observations is

$$\begin{aligned} P(X|Z, \theta) &= \prod_{d=1}^{D} p(X_{:,d}|Z, \theta) \\ &= \prod_{d=1}^{D} \frac{1}{(2\pi)^{N/2}|K|^{1/2}} \exp\left(-\frac{1}{2}X_{:,d}^T K^{-1} X_{:,d}\right). \end{aligned} \quad (10)$$

The likelihood is a product of D independent Gaussian processes, and each process is related to a different dimension of the set of observations. We can reformulate the likelihood as

$$P(X|Z) = \frac{1}{(2\pi)^{N/2}|K|^{1/2}} \exp\left(-\frac{1}{2}\text{tr}(K^{-1}XX^T\right). \quad (11)$$

III. Supervised GP-LVM

In this section, we introduce the supervised extension of the GP-LVM, called the supervised GP-LVM. The supervised GP-LVM is built upon the trick that utilizes a latent variable to connect observations and their corresponding labels. This trick is very popular in statistical models, e.g., joint manifold model [16] and supervised probabilistic PCA (SPPCA) [29].

The supervised GP-LVM is a supervised manifold learning algorithm which can accomplish supervised learning tasks. In the following, an intuitionist example is given to show the efficiency of the label information.

In this example, 200 samples are randomly selected for digits "3" and "5" from the handwritten digit the United States Postal Service (USPS) databases, respectively. The dimensionality reduction results for a total of 400 samples are shown in Fig. 2. The left subfigure is the result obtained by the GP-LVM, and the right subfigure is the result obtained by the supervised GP-LVM. Compared with the left subfigure, digits "3" and "5" can be well separated in the right subfigure. The result of the supervised GP-LVM is superior to the GP-LVM because it considers the label information in the training stage.

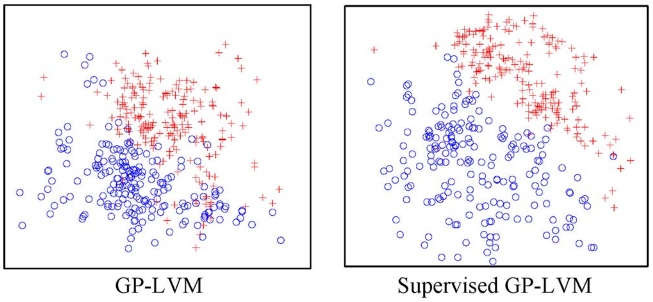


Fig. 2. Dimensionality reduction results for the USPS database. The left subfigure is the result obtained by the GP-LVM, and the right subfigure is the result obtained by the supervised GP-LVM. The supervised GP-LVM is superior to the GP-LVM because it considers the label information in the training stage.

The purpose of the supervised GP-LVM is to find a low-dimensional manifold of a set of high-dimensional observations, which is similar to the GP-LVM. The supervised GP-LVM can duly utilize the sample label information for supervised learning tasks, which is different from the GP-LVM.

The manifold is represented by latent variables. We can establish the relationship between pairs of observations associated with corresponding labels (X, Y) and latent variables Z according to

$$\begin{cases} x = f(z, \theta) \\ y = g(z, \gamma). \end{cases} \quad (12)$$

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

In the supervised GP-LVM, both f and g are Gaussian processes, and latent variables $z \sim N(0, I)$. Then, we can obtain the posterior of z from (X, Y) according to the Bayes' theorem

$$p(Z|X, Y) = \frac{p(X, Y|Z)p(Z)}{p(X, Y)} \quad (13)$$

where $p(X, Y|Z)$ is the likelihood of the pairs of observations and labels (X, Y), and $p(X, Y)$ is the marginal likelihood of all pairs integrated over Z.

The log posterior of $p(Z|X, Y)$ is given by

$$\ln p(Z|X, Y) = \ln p(X, Y|Z) + \ln p(Z) - \ln p(X, Y). \quad (14)$$

The last term in the right-hand side of (13) is the log marginal likelihood, so it is irrelevant to latent variables Z. Maximizing the log posterior $p(Z|X, Y)$ is equivalent to maximizing the likelihood function $p(X, Y|Z)$ plus the prior $p(Z)$

$$\{Z, \theta, \gamma\} = \arg\max_{Z,\theta,\gamma} \{\ln p(Z|X, Y)\}. \quad (15)$$

The solution of (15) is identical to

$$\{Z, \theta, \gamma\} = \arg\max_{X,\theta,\gamma} \{\ln p(X, Y|Z) + \ln p(Z)\}. \quad (16)$$

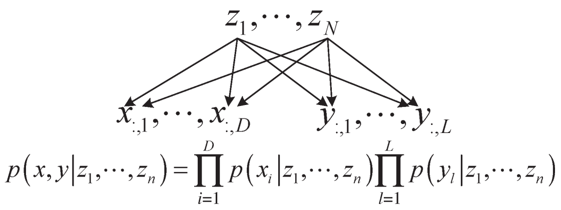

$z_1,\cdots,z_N$
$x_{:,1},\cdots,x_{:,D}$
$y_{:,1},\cdots,y_{:,L}$
$p(x,y|z_1,\cdots,z_n)=\prod_{i=1}^{D}p(x_i|z_1,\cdots,z_n)\prod_{l=1}^{L}p(y_l|z_1,\cdots,z_n)$

Fig. 3. All of the dimensions of the input data x and the output data y are conditionally independent, given latent variable z; $x_{:,d}$ represents the d dimension of all of the input data X; and $y_{:,l}$ represents the l dimension of the output data Y.

By utilizing the conditional independence, observations and corresponding labels are conditionally independent from each other, as shown in Fig. 3.

Then, we have

$$\begin{aligned}\ln p(X,Y|Z) &= \ln p(X|Z) + \ln p(Y|Z)\\ &= \sum_{d=1}^{D}\ln p(X_{:,d}|Z) + \sum_{l=1}^{L}\ln p(Y_{:,l}|Z). \quad (17)\end{aligned}$$

The first part $\sum_{d=1}^{D}\ln p(X_{:,d}|Z)$ in the right-hand side of (17) is the likelihood function of the observations, and the second part $\sum_{l=1}^{L}\ln p(Y_{:,l}|Z)$ can be changed according to different tasks. Fig. 3 shows (17).

Equation (17) contains two terms, and the first term is given by

$$\begin{aligned}L_x &= \ln p(X|Z)\\ &= -\frac{DN}{2}\ln 2\pi - \frac{D}{2}\ln|K| - \frac{1}{2}\mathrm{tr}(K^{-1}XX^T). \quad (18)\end{aligned}$$

The second term in (17) is designed for supervised learning. According to (3), we impose a Gaussian process prior for mapping g. Then, the term is given by

$$\begin{aligned}L_y &= \ln p(Y|Z)\\ &= -\frac{LN}{2}\ln 2\pi - \frac{L}{2}\ln|\Sigma| - \frac{1}{2}\mathrm{tr}(\Sigma^{-1}YY^T). \quad (19)\end{aligned}$$

According to (13)–(19), maximizing Z is equivalent to minimizing

$$L = L_x + L_y + \ln p(Z). \quad (20)$$

In this paper, we apply the scaled conjugate gradient (SCG) with regard to latent variables Z and hyperparameters for training. In particular, we first compute the gradient of (20) with respect to latent variables Z

$$\frac{\partial L}{\partial Z} = \frac{\partial L_x}{\partial Z} + \frac{\partial L_y}{\partial Z} + \frac{\partial \ln p(Z)}{\partial Z}. \quad (21)$$

The gradient of the log likelihood with respect to the latent variable can be computed through the chain rule, i.e.,

$$\frac{\partial L_x}{\partial Z} + \frac{\partial L_y}{\partial Z} = \frac{\partial L_x}{\partial K}\frac{\partial K}{\partial Z} + \frac{\partial L_y}{\partial \Sigma}\frac{\partial \Sigma}{\partial Z} \quad (22)$$

where the gradients of the log likelihood with respect to the kernel matrices are given by

$$\frac{\partial L_x}{\partial K} = K^{-1}XX^TK^{-1} - DK^{-1} \quad (23)$$
$$\frac{\partial L_y}{\partial \Sigma} = \Sigma^{-1}YY^T\Sigma^{-1} - L\Sigma^{-1}. \quad (24)$$

The gradients of the kernel matrices with respect to the latent variables are defined according to the kernel function. If the RBF is selected in the model, then the gradients of the matrices with respect to the latent variable are given by

$$\frac{\partial K}{\partial Z} = \theta_{\mathrm{band}} \cdot K \cdot A \quad (25)$$
$$\frac{\partial \Sigma}{\partial Z} = \gamma_{\mathrm{band}} \cdot \Sigma \cdot A \quad (26)$$

where "·" denotes the dot product between matrices and the ith row and jth column of matrix A are obtained by $z_i - z_j$.

If the prior is defined with $1/2\|Z\|_2$, then the gradient of the prior with respect to Z is equal to

$$\frac{\partial \ln p(Z)}{\partial Z} = \frac{Z}{\|Z\|_2}. \quad (27)$$

The optimal positions of the latent variables can be obtained through the upper process.

Then, we will give the detailed steps for computing the gradient of (20) with respect to hyperparameters θ and γ. Just like the gradient of (20) with respect to the latent variable, we should also find the gradient of (20) with respect to the hyperparameters through the chain rule

$$\begin{cases}\frac{\partial L}{\partial \theta} = \left(\frac{\partial L_x}{\partial K} + \frac{\partial L_y}{\partial K} + \frac{\partial \ln p(Z)}{\partial K}\right)\frac{\partial K}{\partial \theta} = \frac{\partial L_x}{\partial K}\frac{\partial K}{\partial \theta}\\ \frac{\partial L}{\partial \gamma} = \left(\frac{\partial L_x}{\partial \Sigma} + \frac{\partial L_y}{\partial \Sigma} + \frac{\partial \ln p(Z)}{\partial \Sigma}\right)\frac{\partial \Sigma}{\partial \gamma} = \frac{\partial L_y}{\partial \Sigma}\frac{\partial \Sigma}{\partial \gamma}.\end{cases} \quad (28)$$

The gradient of the log likelihood with respect to the kernel matrices has been obtained through (28). Then, the gradients of the kernel matrices with respect to the hyperparameters can be computed by

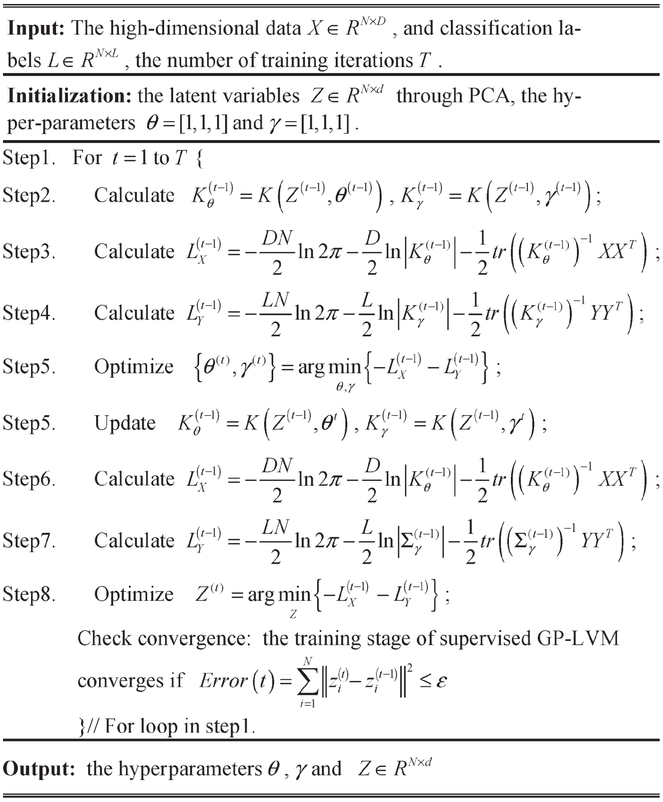
$$\begin{cases}\frac{\partial K}{\partial \theta_{\mathrm{rbf}}} = \frac{1}{\theta_{\mathrm{rbf}}}K\\ \frac{\partial K}{\partial \theta_{\mathrm{band}}} = K \cdot B\\ \frac{\partial K}{\partial \sigma_x^2} = I\end{cases} \quad (29)$$

$$\begin{cases}\frac{\partial \Sigma}{\partial \gamma_{\mathrm{rbf}}} = \frac{1}{\gamma_{\mathrm{rbf}}}\Sigma\\ \frac{\partial \Sigma}{\partial \gamma_{\mathrm{band}}} = \Sigma \cdot B\\ \frac{\partial \Sigma}{\partial \sigma_y^2} = I\end{cases} \quad (30)$$

where $B_{i,j} = -1/2(z_i - z_j)^T(z_i - z_j)$ is the entry in the ith row and jth column of matrix B. Then, the model could be trained by using the SCG.

The detailed steps for the training process will be given in Table I. First, the latent variables will be initialized through the PCA, and all of the values in the hyperparameters are defined as one. Then, the latent variables and the hyperparameters will be optimized by turns. Finally, the supervised GP-LVM can be established with the hyperparameters obtained from the training process.

TABLE I
TRAINING PROCESS FOR THE SUPERVISED GP-LVM



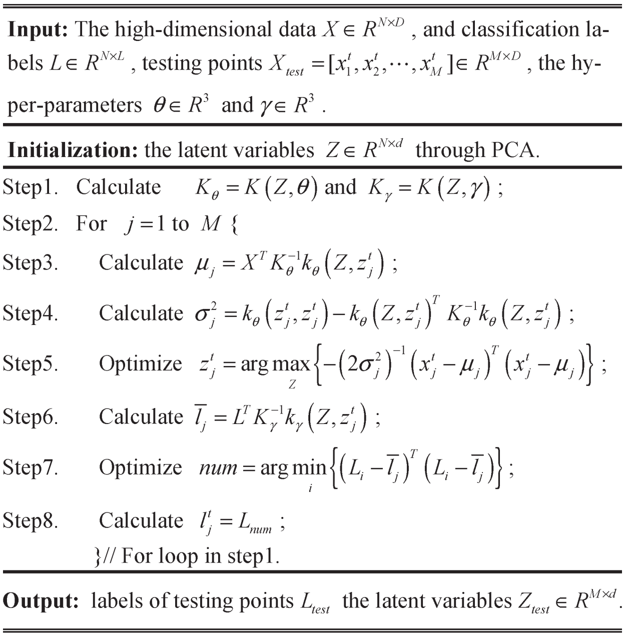
Input: The high-dimensional data $X \in R^{N \times D}$, and classification labels $L \in R^{N \times L}$, the number of training iterations T.

Initialization: the latent variables $Z \in R^{N \times d}$ through PCA, the hyper-parameters $\theta = [1,1,1]$ and $\gamma = [1,1,1]$.

```
Step1.  For $t = 1$ to $T$ {
Step2.      Calculate $K_\theta^{(t-1)} = K\left(Z^{(t-1)}, \theta^{(t-1)}\right)$, $K_\gamma^{(t-1)} = K\left(Z^{(t-1)}, \gamma^{(t-1)}\right)$;
Step3.      Calculate $L_X^{(t-1)} = -\frac{DN}{2}\ln 2\pi - \frac{D}{2}\ln\left|K_\theta^{(t-1)}\right| - \frac{1}{2}tr\left(\left(K_\theta^{(t-1)}\right)^{-1} XX^T\right)$;
Step4.      Calculate $L_Y^{(t-1)} = -\frac{LN}{2}\ln 2\pi - \frac{L}{2}\ln\left|K_\gamma^{(t-1)}\right| - \frac{1}{2}tr\left(\left(K_\gamma^{(t-1)}\right)^{-1} YY^T\right)$;
Step5.      Optimize $\left\{\theta^{(t)}, \gamma^{(t)}\right\} = \arg\min_{\theta,\gamma}\left\{-L_X^{(t-1)} - L_Y^{(t-1)}\right\}$;
Step5.      Update $K_\theta^{(t-1)} = K\left(Z^{(t-1)}, \theta^t\right)$, $K_\gamma^{(t-1)} = K\left(Z^{(t-1)}, \gamma^t\right)$;
Step6.      Calculate $L_X^{(t-1)} = -\frac{DN}{2}\ln 2\pi - \frac{D}{2}\ln\left|K_\theta^{(t-1)}\right| - \frac{1}{2}tr\left(\left(K_\theta^{(t-1)}\right)^{-1} XX^T\right)$;
Step7.      Calculate $L_Y^{(t-1)} = -\frac{LN}{2}\ln 2\pi - \frac{L}{2}\ln\left|\Sigma_\gamma^{(t-1)}\right| - \frac{1}{2}tr\left(\left(\Sigma_\gamma^{(t-1)}\right)^{-1} YY^T\right)$;
Step8.      Optimize $Z^{(t)} = \arg\min_{Z}\left\{-L_X^{(t-1)} - L_Y^{(t-1)}\right\}$;
            Check convergence: the training stage of supervised GP-LVM
            converges if $Error(t) = \sum_{i=1}^{N}\left\|z_i^{(t)} - z_i^{(t-1)}\right\|^2 \le \varepsilon$
        }// For loop in step1.
```

Output: the hyperparameters θ, γ and $Z \in R^{N \times d}$

TABLE II
TESTING PROCESS FOR THE SUPERVISED GP-LVM



Input: The high-dimensional data $X \in R^{N \times D}$, and classification labels $L \in R^{N \times L}$, testing points $X_{test} = [x_1^t, x_2^t, \cdots, x_M^t] \in R^{M \times D}$, the hyper-parameters $\theta \in R^3$ and $\gamma \in R^3$.

Initialization: the latent variables $Z \in R^{N \times d}$ through PCA.

```
Step1.  Calculate $K_\theta = K(Z, \theta)$ and $K_\gamma = K(Z, \gamma)$;
Step2.  For $j = 1$ to $M$ {
Step3.      Calculate $\mu_j = X^T K_\theta^{-1} k_\theta\left(Z, z_j^t\right)$;
Step4.      Calculate $\sigma_j^2 = k_\theta\left(z_j^t, z_j^t\right) - k_\theta\left(Z, z_j^t\right)^T K_\theta^{-1} k_\theta\left(Z, z_j^t\right)$;
Step5.      Optimize $z_j^t = \arg\max_{Z}\left\{-\left(2\sigma_j^2\right)^{-1}\left(x_j^t - \mu_j\right)^T\left(x_j^t - \mu_j\right)\right\}$;
Step6.      Calculate $\overline{l}_j = L^T K_\gamma^{-1} k_\gamma\left(Z, z_j^t\right)$;
Step7.      Optimize $num = \arg\min_{i}\left\{\left(L_i - \overline{l}_j\right)^T\left(L_i - \overline{l}_j\right)\right\}$;
Step8.      Calculate $l_j^t = L_{num}$;
        }// For loop in step1.
```

Output: labels of testing points L_{test} the latent variables $Z_{test} \in R^{M \times d}$.

Through the training process, hyperparameters θ and γ can be obtained by maximizing (20). For the test samples $X_{\text{test}} \in R^{M \times D}$, we estimate latent variables Z_{test} that correspond to test points X_{test} with the estimated hyperparameters θ. According to the study in [21], the point can be shown as a Gaussian distribution after projection

$$p\left(x_j^t | z_j^t\right) = N\left(x_j^t | \mu_j, \sigma_j^2 \cdot I\right) \tag{31}$$

where $\mu_j = X^T K_\theta^{-1} k(Z, z_j^t)$ is the mean of the Gaussian distribution and $\sigma_j^2 = k(z_j^t, z_j^t) - k(Z, z_j^t)^T K_\theta^{-1} k(Z, z_j^t)$ is the variance.

We can obtain z_j^t by optimizing (32) by using the SCGs. First, the gradients of (31) with respect to latent variable z_j^t can be obtained through the chain rule

$$\frac{\partial p\left(x_j^t | z_j^t\right)}{\partial z_j^t} = \frac{\partial p\left(x_j^t | z_j^t\right)}{\partial \mu_j}\frac{\partial \mu_j}{\partial z_j^t} + \frac{\partial p\left(x_j^t | z_j^t\right)}{\partial \sigma_j^2}\frac{\partial \sigma_j^2}{\partial z_j^t}. \tag{32}$$

Then, the first term in the right-hand side is divided into two parts, i.e., $\partial p(x_j^t | z_j^t)/\partial \mu_j$ and $\partial \mu_j / \partial z_j^t$. The former is the gradient of a Gaussian distribution with respect to the mean value μ_j. The latter is the gradient of the mean value with respect to the latent variable

$$\frac{\partial \mu_j}{\partial z_j^t} = -X^T K_\theta^{-1}\left(k\left(Z, z_j^t\right) \cdot \frac{\theta_{\text{band}}}{2}\left(Z - z_j^t\right)\right). \tag{33}$$

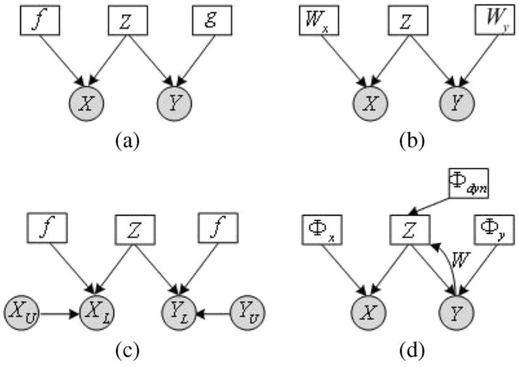


Fig. 4. Graphical representations of the (a) proposed supervised GP-LVM, (b) SPPCA [19], (c) joint manifold model [16], and (d) dynamic shared LVM [17].

The second term in the right-hand side of (33) can also be computed in two parts $\partial p(x_j^t | z_j^t)/\partial \sigma_j^2$ and $\partial \sigma_j^2 / \partial z_j^t$. The former is the gradient of a Gaussian distribution with respect to the variance, and the latter is the gradient of the mean value with respect to the latent variable

$$\frac{\partial \sigma_j^2}{\partial z_j^t} = 2K_\theta^{-1}\left(k\left(Z, z_j^t\right) \cdot \frac{\theta_{\text{band}}}{2}\left(Z - z_j^t\right)\right). \tag{34}$$

Finally, the position of the test variable x_j^t in the latent space can be optimized with the SCG methods.

After that, we can compute the label l_j^t for the test sample z_j^t with the hyperparameters γ. Table II shows the prediction procedure.

A. Discussions

Fig. 4 shows the graphical representations of the proposed supervised GP-LVM, the SPPCA [19], the joint manifold model [16], and the dynamic shared LVM [17] Based on this figure, we can understand that these models are intrinsically different from each other.

1) The supervised GP-LVM: the new model is a nonlinear supervised dimensionality reduction model. The model of the label information is different from the model of the observed data set X that is used to achieve an effective and efficient dimension reduction. By treating the observed data set X and the label set Y differently, the supervised GP-LVM uses two Gaussian processes with different hyperparameters and shared latent variables Z. The mapping functions are

$$\begin{cases} X = f(Z, \theta) \\ Y = g(Z, \gamma). \end{cases} \tag{35}$$

The corresponding objective function is the posterior of latent variable Z, i.e.,

$$P(Z|X, Y) = \frac{P(X|f, Z)P(Y|g, Z)p(Z)}{P(X, Y)}. \tag{36}$$

2) The SPPCA [19]: it is a supervised extension of the probabilistic PCA, which is achieved by introducing the linear mapping functions W_x and W_y, as shown in Fig. 4(b). By considering the label information as the output, the SPPCA links the observed data set X and the label set Y via latent variables Z

$$\begin{cases} X = W_x Z + \mu_x + \varepsilon_{wx} \\ Y = W_y Z + \mu_y + \varepsilon_{wy}. \end{cases} \tag{37}$$

The parameters are obtained by maximizing the log likelihood

$$L_{\mathrm{SPPCA}} = \sum_{n=1}^{N} \log \int p(\mathrm{x}_n|z_n) p(\mathrm{y}_n|z_n) p(\mathrm{z}_n) d\mathrm{z}_n. \tag{38}$$

3) The joint manifold model [16]: in order to utilize the unlabeled observed samples, a semisupervised regression learning model is proposed based on the LVM [16]. The model offers a Gaussian process from latent variable Z to the observed data set, including both labeled samples X_L and Y_L and unlabeled samples X_u and Y_u. If we let X be $X_L \cup X_u$ and Y be $Y_L \cup Y_u$, we have

$$\begin{cases} X = f(Z) \\ Y = f(Z). \end{cases} \tag{39}$$

The objective function is the joint likelihood of the observed data sets, i.e.,

$$P(X, Y|Z) = P(X|f, Z)P(Y|f, Z). \tag{40}$$

4) The dynamic shared LVM [17]: it aims to learn a dynamical model over the latent space, which allows us to disambiguate between ambiguous silhouettes by temporal consistency. Fig. 4(d) shows that the model combines the shared LVM with the dynamical model for human pose estimation. The objective function of the study in [17] is a joint likelihood

$$P(X, Y, W|\Phi) = P(X|\Phi_X, W)P(Y|\Phi_Y, W)P(W|\Phi_{\mathrm{dyn}}) \tag{41}$$

where Φ_X and Φ_Y denote the mapping functions between the observed samples X and labels Y, respectively, and Φ_{dyn} and W represent the dynamic functions in the latent space.

IV. Experimental Results

In this section, we first visualize the latent variables in a 2-D space by using the GP-LVM, the GDA, the discriminative GP-LVM, and the proposed supervised GP-LVM. We also conduct experiments on some University of California Irvine (UCI) wine data sets [36] and on the USPS database [21]. Finally, the comparison with some supervised classifiers, such as Gaussian process classification [Laplace approximation (LA) and expectation propagation (EP)] [25] and support vector machines (SVMs) [37], is conducted. The experimental results show that the supervised GP-LVM is superior compared to the GP-LVM, the GDA, and the discriminative GP-LVM in terms of effectiveness.

A. Data Visualization

1) Oil Flow Data: We use the iris data [36] and oil data set [20] to verify the effectiveness of the proposed supervised GP-LVM in dealing with dimensionality reduction for data visualization. Figs. 5 and 6 show the results of the following four dimensionality reduction algorithms: the GP-LVM, the GDA, the discriminative GP-LVM [28], and the proposed one. The four subfigures in Figs. 5 and 6 show the results of the following four models: 1) the GP-LVM, 2) the GDA, 3) the discriminative GP-LVM, and 4) the proposed supervised GP-LVM. The discriminative GP-LVM can be interpreted as a regularized GP-LVM. It discovers a tradeoff between the GP-LVM and the GDA by changing a tuning parameter σ_d^2. In particular, $\sigma_d^2 = 1$ for 3), $\sigma_d^2 = \infty$ for 1), and $\sigma_d^2 = \infty$ for 2).

The visualizations of the *oil* database are shown in Fig. 5. This database contains three classes with 1000 samples in R^{12}.

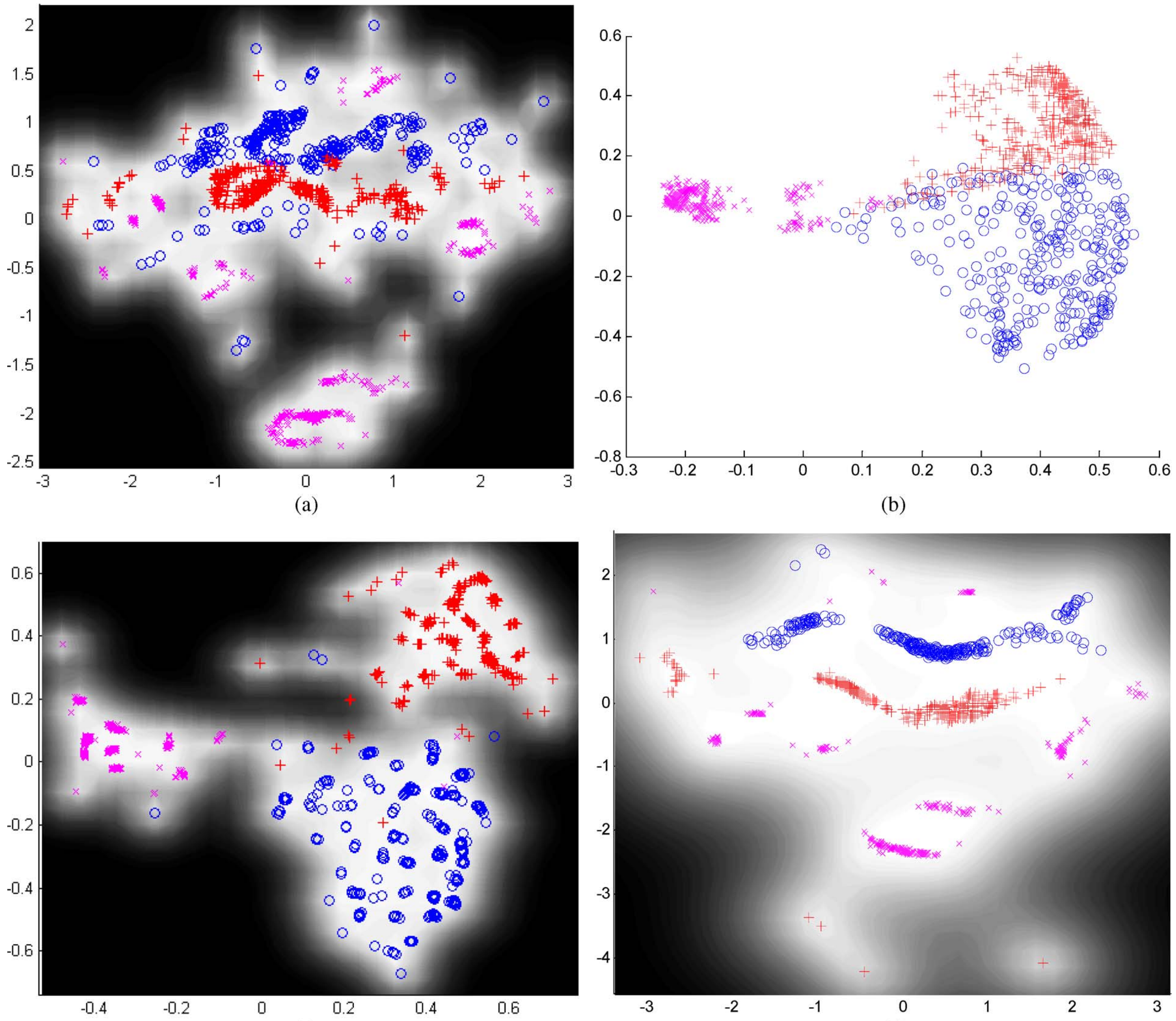
2) Iris Data: The *iris* data set contains 50 samples from each of the three species of iris flowers. Four features were measured for each sample, which are the length and width of the sepal and petal.

By comparing four subfigures in Figs. 5 and 6, for the supervised GP-LVM, samples in the same class are distributed tightly, and compact manifolds can be obtained for each class. In addition, samples in different classes can be well separated in comparing with the other three algorithms.

B. Classification Comparison With Dimensionality Reduction Methods in the Latent Variable Space

Figs. 7 and 8 show that the classification error rates change with the size of the training set. The three aforementioned methods are compared with the proposed supervised GP-LVM. For all of the five methods, the nearest-neighbor-based classification is introduced in the learned latent variable space. Figs. 7 and 8 show the mean error rates of the *USPS* [21] and *wine* databases [36], respectively.

1) USPS Database: The experimental results of the *USPS* test are shown in Fig. 7. The classification errors decrease with the increase of the number of the training samples.



(a)

(b)

(c)

(d)

Fig. 5. Full oil flow data set visualized with (a) GP-LVM, (b) GDA, (c) discriminative GP-LVM, and (d) supervised GP-LVM.

We tested five algorithms on digits 3 and 5. For each digit, 500 images are selected, and each image is in R^{256}. In the following experiments, the dimension is reduced to one, and 20–160 samples (with a step of 20) are chosen randomly from each digit for training (with the rest for testing).

2) Wine Data Set: This data set contains three classes, with a total of 178 samples. The dimension of each sample is 13.

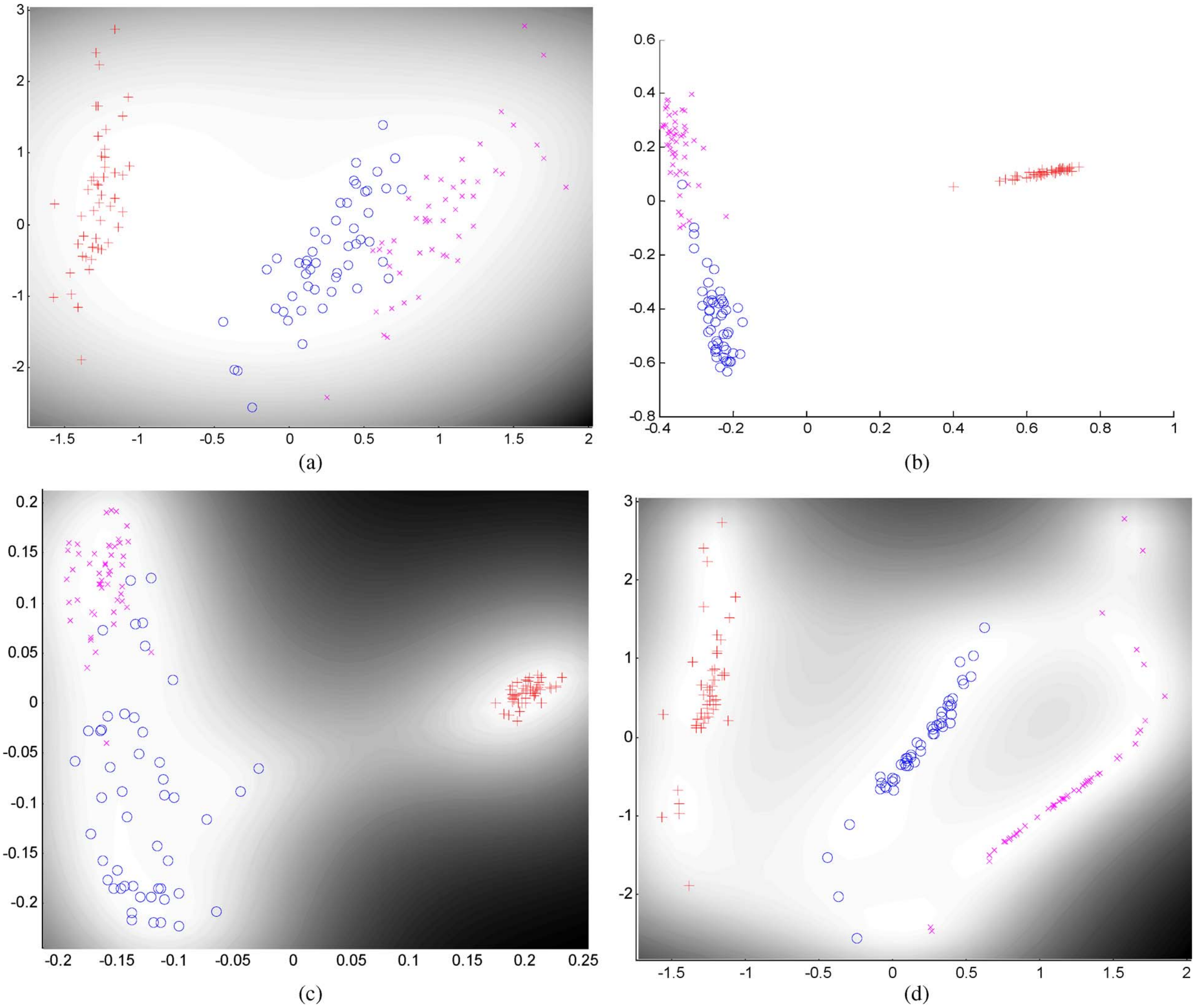
Fig. 7 shows the results of the supervised GP-LVM in comparison with the other four aforementioned models. In this experiment, the dimension of the latent variables is one. Fig. 8 shows the classification accuracies of the UCI *wine* database. In this experiment, the dimension of the examples is reduced to two. For both experiments, the supervised GP-LVM performs better than the others. The linear discriminant analysis (LDA) can reach lower error rates compared to the GP-LVM, discriminative GP-LVM and GDA, whereas it does not achieve better results than the proposed method. The discriminative GP-LVM performed comparatively to GDA when the size of the training set is small. Moreover, the mean error of the GP-LVM is much higher than the others because it ignores the sample label information.

C. Classification Comparison With Supervised Classifiers

This section evaluates the performance of the proposed method in comparison with three typical supervised classifiers (i.e., LA, EP, and SVMs) on two UCI databases, including *ionosphere* [36] and *sonar* [36]. LA and EP are two traditional methods used in Gaussian process classification. For the proposed supervised GP-LVM, we choose two different dimensions of the latent space for each data set. The dimensions of the latent space are chosen as $d = 4$ and $d = 10$ for the *ionosphere* data set, and the dimensions of the latent space are $d = 4$ and $d = 15$ for the *sonar* data set. The three reference classification models are performed in the original space. Each data set can be divided into two parts; one is a training set whose samples are randomly selected from the data sets, and the remaining part is used for testing.

1) Ionosphere Data Set: The *ionosphere* data set contains two classes, with 351 samples in R^{34}.

2) Sonar Data Set: The *sonar* data set contains 208 samples in R^{60}. The samples are classified into two classes: “rock” and “mine.”



(a) (b)

(c) (d)

Fig. 6. Iris data set visualized with (a) GP-LVM, (b) GDA, (c) discriminative GP-LVM, and (d) supervised GP-LVM.

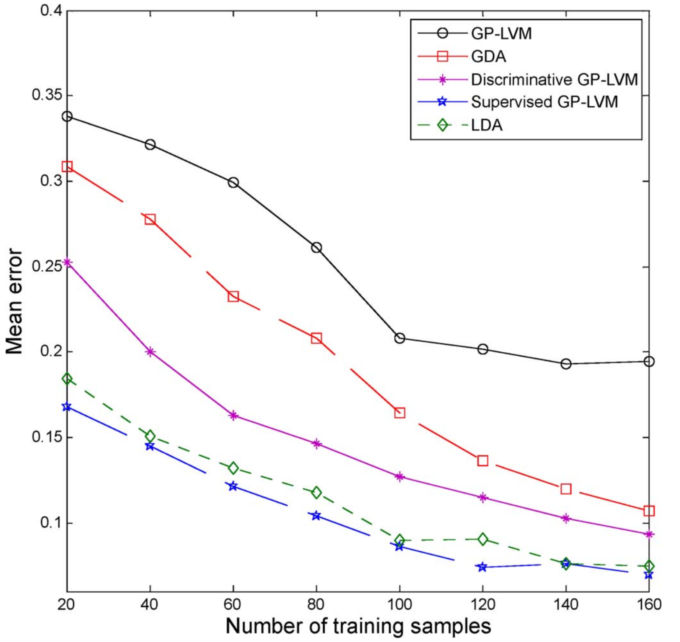


Fig. 7. Mean error rates of the five methods change with the number of training samples for the USPS data sets.

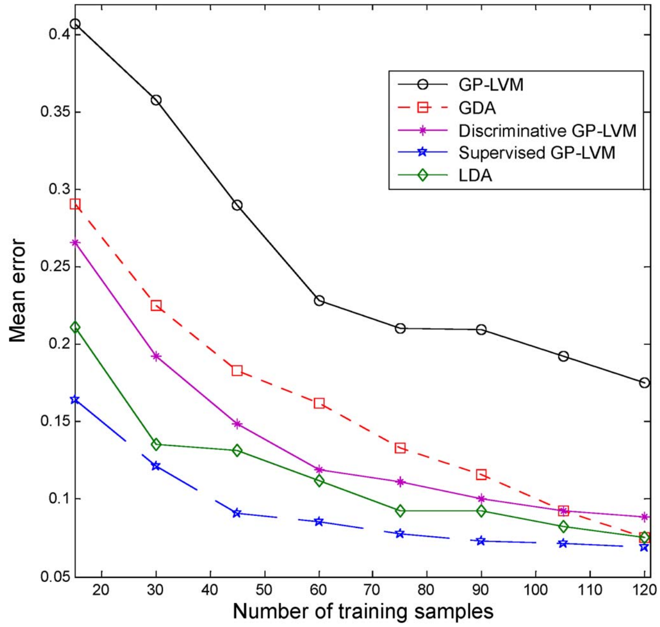


Fig. 8. Mean error rates of the five methods change with the number of training samples for the UCI wine data set.

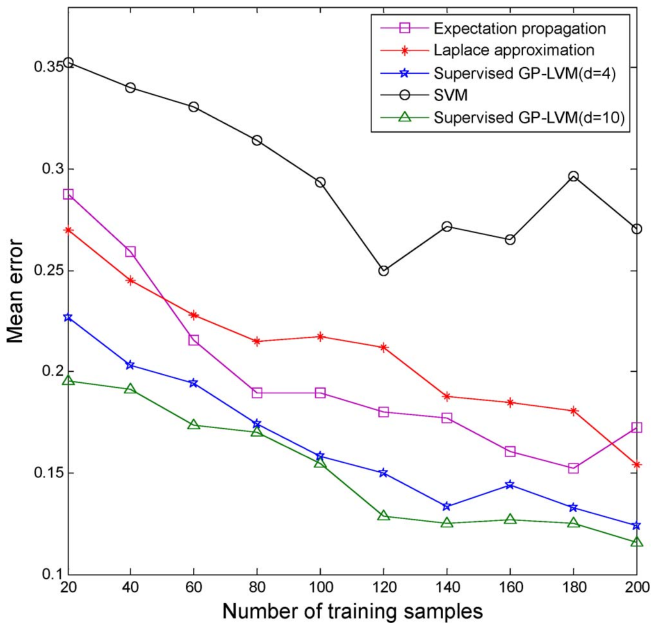


Fig. 9. Mean error rates of the four methods change with the number of training samples for the UCI ionosphere data set.

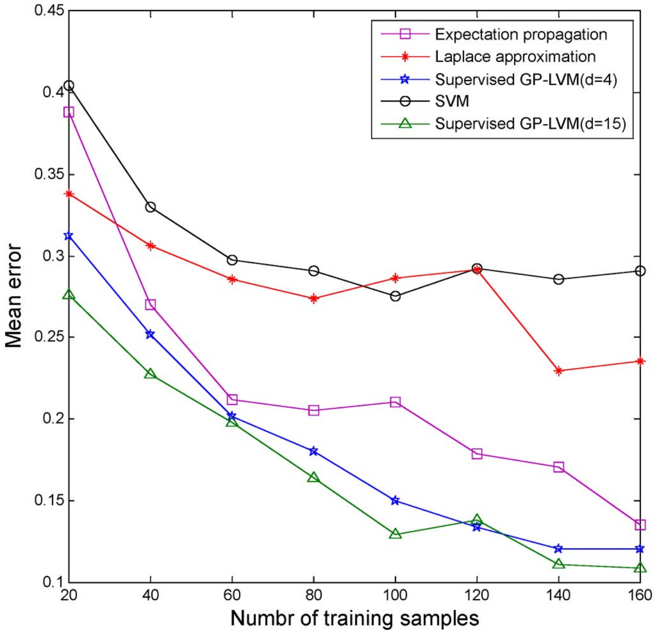


Fig. 10. Mean error rates of the four methods change with the number of training samples for the UCI sonar data set.

Figs. 9 and 10 show the misclassification rates versus the number of training samples for four different classification algorithms. The kernel functions used in the aforementioned four methods are nonlinear RBFs. The experimental results show that the proposed method consistently and significantly outperforms the other three classifiers on two data sets because the proposed method includes a dimensionality reduction process that is used to eliminate the irrelevant and redundant information in the data sets. However, the three reference methods do not include any dimensionality reduction process, and they directly conduct classification in the original space. Therefore, the misclassification rates would be higher than the proposed method.

V. Conclusion

In this paper, the supervised GP-LVM has been developed for supervised learning tasks, and then, the maximum *a posteriori* algorithm is introduced to estimate the latent variables. One of the most important advantages of the supervised GP-LVM, compared to the GP-LVM, is that the joint likelihood is learned not only for the observed samples but also for the labels. The empirical studies on data visualization and classification show that the supervised GP-LVM outperforms the GP-LVM, the GDA, and the discriminative GP-LVM.

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

[1] H. Hotelling, “Analysis of a complex of statistical variables into principal components,” *J. Educational Psychol.*, vol. 24, no. 7, pp. 417–441, Oct. 1933.

[2] T. Zhou, D. Tao, and X. Wu, “Manifold elastic net: A unified framework for sparse dimension reduction,” *Data Mining Knowl. Discovery*, 2010, DOI: 10.1007/s10618-010-0182-x.

[3] Y. Pang, Y. Yuan, and X. Li, “Iterative subspace analysis based on feature line distance,” *IEEE Trans. Image Process.*, vol. 18, no. 4, pp. 903–907, Apr. 2009.

[4] S. Si, D. Tao, and K. P. Chan, “Evolutionary cross-domain discriminative Hessian eigenmaps,” *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 1075–1086, Apr. 2010.

[5] T. Zhang, D. Tao, X. Li, and J. Yang, “Patch alignment for dimensionality reduction,” *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1299–1313, Sep. 2009.

[6] X. He and P. Niyogi, “Locality preserving projections,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 153–160.

[7] X. He, D. Cai, and S. Yan, “Neighborhood preserving embedding,” in *Proc. 10th Int. Conf. Comput. Vis.*, 2005, pp. 1208–1213.

[8] W. Bian and D. Tao, “Biased discriminant Euclidean embedding for content based image retrieval,” *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 545–554, Feb. 2010.

[9] D. Song and D. Tao, “Biologically inspired feature manifold for scene classification,” *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 174–184, Jan. 2010.

[10] D. Cai, X. He, and J. W. Han, “Orthogonal Laplacianfaces for face recognition,” *IEEE Trans. Image Process.*, vol. 15, no. 11, pp. 3608–3614, Nov. 2006.

[11] Y. Yuan, X. Li, Y. Pang, X. Lu, and D. Tao, “Binary sparse nonnegative matrix factorization,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 5, pp. 772–777, May 2009.

[12] X. Li, Y. Pang, and Y. Yuan, “L1-norm-based 2DPCA,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 4, pp. 1170–1175, Aug. 2010.

[13] X. Li, S. Lin, S. Yan, and D. Xu, “Discriminant locally linear embedding with high-order tensor data,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 2, pp. 342–352, Apr. 2008.

[14] D. J. Bartholomew, *Statistical Factor Analysis and Related Methods*. New York: Wiley, 2004.

[15] M. E. Tipping and C. M. Bishop, “Probabilistic principal component analysis,” *J. Roy. Statist. Soc., B*, vol. 61, no. 3, pp. 611–622, 1999.

[16] S. T. Roweis, “EM algorithms for PCA and SPCA,” in *Proc. Adv. Neural Inf. Process. Syst.*, 1997, pp. 626–632.

[17] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K.-R. Muller, “Fisher discriminant analysis with kernels,” in *Proc. IEEE Neural Netw. Signal Process. Workshop IX*, 1999, pp. 41–48.

[18] S. Si, D. Tao, and B. Geng, “Bregman divergence based regularization for transfer subspace learning,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 7, pp. 929–942, Jul. 2010.

[19] X. Li and Y. Pang, “Deterministic column-based matrix decomposition,” *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 1, pp. 145–149, Jan. 2010.

[20] B. Scholkopf, A. Smola, and K.-R. Muller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.

[21] N. D. Lawrence, “Probabilistic non-linear principal component analysis with Gaussian process latent variable models,” *J. Mach. Learn. Res.*, vol. 6, pp. 1783–1816, Nov. 2005.

[22] J. M. Wang, D. J. Fleet, and A. Hertzmann, “Gaussian process dynamical models for human motion,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 283–289, Feb. 2008.

[23] X. Wang, X. Gao, Y. Yuan, and D. Tao, “Semi-supervised Gaussian process latent variable model with pairwise constraints,” *Neurocomputing*, vol. 73, no. 10–12, pp. 2186–2195, Jun. 2010.

[24] X. Gao, X. Wang, X. Li, and D. Tao, “Transfer latent variable model based on divergence analysis,” *Pattern Recognit.*, 2010, DOI: 10.1016/j.patcog.2010.06.013.

[25] C. E. Rasmussen and C. K. I. Williams, *Gaussian Process for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[26] R. Navaratnam, A. W. Fitzgibbon, and R. Cipolla, “The joint manifold model for semi-supervised multi-valued regression,” in *Proc. 8th Int. Conf. Comput. Vis.*, 2007, pp. 1–8.

[27] C. Henrik, E. P. Torr, and N. D. Lawrence, “Gaussian process latent variable models for human pose estimation,” in *Proc. Mach. Learn. Multimodal Interaction*, 2008, pp. 132–143.

[28] R. Urtasun and T. Darrell, “Discriminative Gaussian process latent variable model for classification,” in *Proc. 24th Int. Conf. Mach. Learn.*, 2007, pp. 927–934.

[29] S. P. Yu, K. Yu, V. Tresp, H.-P. Kriegel, and M. Wu, “Supervised probabilistic principal component analysis,” in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2006, pp. 464–473.

[30] A. P. Dawid, “Conditional independence in statistical theory,” *J. Roy. Statist. Soc., B*, vol. 41, no. 1, pp. 1–31, 1979.

[31] D. J. Bartholomew, *Latent Variable Models and Factor Analysis*. London, U.K.: Griffin, 1987.

[32] C. K. I. Williams and C. E. Rasmussen, “Gaussian processes for regression,” in *Proc. Adv. Neural Inf. Process. Syst.*, 1996, pp. 598–604.

[33] C. K. I. Williams and D. Barber, “Bayesian classification with Gaussian processes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1342–1351, Dec. 1998.

[34] G. Baudat and F. Anouar, “Generalized discriminant analysis using a kernel approach,” *Neural Comput.*, vol. 12, no. 10, pp. 2385–2404, Oct. 2000.

[35] R. A. Fisher, “The statistical utilization of multiple measurements,” *Ann. Eugenics*, vol. 8, pp. 376–386, 1938.

[36] [Online]. Available: http://archive.ics.uci.edu/ml/datasets.html

[37] C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” *Data Mining Knowl. Discovery*, vol. 2, no. 2, pp. 121–167, Jun. 1998.

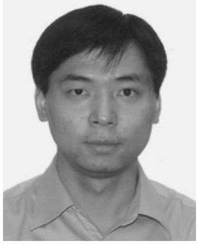
Xinbo Gao (M’02–SM’07) received the B.Sc., M.Sc., and Ph.D. degrees in signal and information processing from Xidian University, Xi’an, China, in 1994, 1997, and 1999, respectively.

From 1997 to 1998, he was a Research Fellow with the Department of Computer Science, Shizuoka University, Shizuoka, Japan. From 2000 to 2001, he was a Postdoctoral Research Fellow with the Department of Information Engineering, Chinese University of Hong Kong, Shatin, Hong Kong. Since 2001, he has been with the School of Electronic Engineering, Xidian University. He is currently a Professor of pattern recognition and intelligent system and the Director of the VIPS Lab, Xidian University. He has published four books and around 100 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 Neural Networks, IEEE Transactions on Systems, Man, and Cybernetics, etc. He is on the editorial boards of international journals, including EURASIP’s *Signal Processing* (Elsevier) and *Neurocomputing* (Elsevier). His research interests include machine learning, computational intelligence, pattern recognition, and video content analysis.

Dr. Gao served as the General Chair/Cochair or Program Committee (PC) Chair/Cochair or PC member for around 30 major international conferences.

Xiumei Wang received the B.Math. degree from Shandong Normal University, Jinan, China, in 2002 and the M.Sc. degree from Xidian University, Xi’an, China, in 2005. She is currently working toward the Ph.D. degree in the School of Electronic Engineering, Xidian University.

Her research interests mainly involve nonparametric statistical models and machine learning.



Dacheng Tao (M’07) received the Ph.D. degree from the University of London, London, U.K.

He is currently a Nanyang Assistant Professor with the Nanyang Technological University, Singapore; a Research Associate Fellow with the University of London; a Visiting Professor with Xidian University, Xi’an, China; and a Guest Professor with Wuhan University, Wuhan, China. He has authored more than 150 scientific articles at top venues, including IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Knowledge and Data Engineering, the Neural Information Processing Systems, ACM SIGKDD Conference on Knowledge Discovery and Data Mining, International Conference on Data Mining, and the conference on Artificial Intelligence and Statistics. He is an Associate Editor for Elsevier’s *Neurocomputing*. He works on computational neuroscience, biologically inspired models, statistics, and their applications in computational vision and video surveillance.

Dr. Tao is a member of the IEEE Technical Committee on Cognitive Computing. He is an Associate Editor of the IEEE Transactions on Knowledge and Data Engineering. He has (co)chaired more than 30 times for special sessions, invited sessions, workshops, panels, and conferences. He has served on more than 110 major international conferences, including International Conference on Data Mining, ACM SIGKDD Conference on Knowledge Discovery and Data Mining, IEEE Conference on Computer Vision and Pattern Recognition, International Conference on Computer Vision, and European Conference on Computer Vision, and on more than 50 prestigious international journals. He received numerous best paper awards.

Xuelong Li (M’02–SM’07) is a Researcher (Full Professor) with the State Key Laboratory of Transient Optics and Photonics and the director of the Center for **OPT**ical **IM**agery **A**nalysis and **L**earning (OPTIMAL), Xi’an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi’an, Shaanxi, China.