Bayesian Maximum Margin Principal Component Analysis

Changying Du[1,2], **Shandian Zhe**[3], **Fuzhen Zhuang**[1], **Yuan Qi**[3], **Qing He**[1], **Zhongzhi Shi**[1]
[1]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing 100190, China
[2]University of Chinese Academy of Sciences, Beijing 100049, China
[3]Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA
{ducy, zhuangfz, heq, shizz}@ics.ict.ac.cn, {alanqi, szhe}@cs.purdue.edu

Abstract

Supervised dimensionality reduction has shown great advantages in finding predictive subspaces. Previous methods rarely consider the popular maximum margin principle and are prone to overfitting to usually small training data, especially for those under the maximum likelihood framework. In this paper, we present a posterior-regularized Bayesian approach to combine Principal Component Analysis (PCA) with the max-margin learning. Based on the data augmentation idea for max-margin learning and the probabilistic interpretation of PCA, our method can automatically infer the weight and penalty parameter of max-margin learning machine, while finding the most appropriate PCA subspace simultaneously under the Bayesian framework. We develop a fast mean-field variational inference algorithm to approximate the posterior. Experimental results on various classification tasks show that our method outperforms a number of competitors.

Introduction

Principal Component Analysis (PCA) has been widely used for dimensionality reduction and data analysis. It aims to extract dominant patterns underlying the data, and to represent it as a set of new orthogonal variables called principal components. Due to its restrictive linear algebra based unsupervised formulation, there have been many efforts to extend this fundamental technique to more general scenarios. Among these work, the probabilistic PCA (PPCA) (Tipping and Bishop 1999) is a prominent example, which allows us to integrate PCA as a bottom layer module into more powerful hierarchical Bayesian frameworks.

Meanwhile, extending PCA with supervised information is another promising direction since this can help to learn more discriminative features for classification and regression analysis. The supervised PPCA model (Yu et al. 2006) extends PPCA by assuming that the Gaussian features and labels are generated independently from a latent low dimensional space through linear transformations. A more general exponential family supervised PCA model proposed in (Guo 2009) assumes each data and label pair is generated from a common latent variable via conditional exponential family models, and optimizes the conditional likelihood of observation pairs via a convex formulation.

Apart from supervised PCA, there also exist many other supervised dimensionality reduction (SDR) methods. The support vector decomposition machine (SVDM) (Pereira and Gordon 2006) uses Singular Value Decomposition (SVD) to find the low dimensional space, while training linear classification models with hinge loss in that space. Later, a more efficient approach based on generalized linear models was proposed in (Rish et al. 2008), which uses a closed-form EM-style procedure to optimize the weighted linear combination of the conditional likelihoods on features and labels. In (Chen et al. 2012), a large-margin harmonium model (MMH) based on latent Markov network was proposed for multi-view data analysis. Recently, Zhu et al. proposed a infinite latent SVM (iLSVM) (Zhu, Chen, and Xing 2014) based on the Indian buffet process (IBP) (Ghahramani and Griffiths 2005), which can infer the most appropriate number of features. However, MMH and iLSVM have to solve many SVM subproblems during their inference procedure, thus tend to be inefficient for large data.

In this paper, we propose a data augmentation based Bayesian posterior regularization approach to combine max-margin learning with PPCA. Unlike MMH and iLSVM, which are both under the maximum entropy discrimination (Jaakkola, Meila, and Jebara 1999) framework, and cannot infer the penalty parameter of max-margin models in a Bayesian style, our method is based on the data augmentation idea for max-margin learning (Polson and Scott 2011), which allows us to automatically infer the weight and penalty parameter while finding the most appropriate PCA subspace simultaneously under the Bayesian framework. Our approach also differs from SVDM, which imposes strict constraints to keep the L2-norm of its model weights always smaller than 1. Finally, compared with maximum likelihood based methods, our Bayesian model has many inherent advantages, e.g., suppressing unnecessary principal components with the automatic relevance determination (Neal 1995; Tipping 2001) prior, and avoiding overfitting to small training set by model averaging, etc.

We apply our general framework to the max-margin classification problem in latent PCA space. To allow our model to scale to large data sets, we develop a fast mean-field variational inference algorithm to approximate the posterior. Experiments on synthetic and real classification tasks show that our method outperforms a number of competitors.

Related Work SDR has been active for a long time (Fukumizu, Bach, and Jordan 2004; Zhang, Zhou, and Chen 2007). Recently, Xu et al. studied SDR in a weakly supervised setting (Xu et al. 2014). Their large margin framework simultaneously encourages angle consistency between preference pairs and maximizes the distance between examples in preference pairs. In (Raeder et al. 2013), Raeder et al. proposed a scalable SDR model with hierarchical clustering. To collapse related features into a single dimension, they cluster model parameters from historical models and implicitly incorporate feature and label data without operating directly in a massive space. In (Mcauliffe and Blei 2008), focusing on text data, a supervised topic model is proposed, which can discover more predictive latent topical representations. Later, Zhu et al. extended it with Bayesian posterior regularization for max-margin learning (Zhu, Ahmed, and Xing 2012; Zhu et al. 2014). But their models are restricted to discrete data while our PCA based model are more general.

Bayesian Maximum Margin PCA

In this section, we first review the probabilistic PCA, and then present the proposed Bayesian max-margin PCA framework. We exemplify it by giving a classification model and a fast variational inference procedure to approximate the posterior. We assume we have a data matrix $\mathbf{X} \in \mathbb{R}^{d\times N}$ consisting of N observations $\{\mathbf{x}_n\}_{n=1}^N$ in d-dimensional feature space. For supervised learning, we also have a $1 \times N$ response vector $\mathbf{y}$.

Probabilistic PCA

The probabilistic PCA (PPCA) (Tipping and Bishop 1999) is a latent variable model, which defines a generative process for each observation $\mathbf{x}$ as

$$\mathbf{x}|\mathbf{z} \sim \mathcal{N}(\mathbf{x}|\mathbf{W}\mathbf{z}+\mathbf{t},\ \sigma^2\mathbf{I}_d), \quad \mathbf{z} \sim \mathcal{N}(\mathbf{z}|\mathbf{0},\ \mathbf{I}_k),$$

where $\mathcal{N}(\cdot)$ is the multivariate normal distribution, $\mathbf{W} \in \mathbb{R}^{d\times k}$ is the factor loading matrix, $\mathbf{z} \in \mathbb{R}^{k\times 1}$ is a k-dimensional latent variable, and $\mathbf{t}$ is a d-dimensional vector which allows non-zero means for the data. Then it is easy to verify that the marginal distribution of observation $\mathbf{x}$ also is a Gaussian, with mean vector $\mathbf{t}$ and covariance matrix $\mathbf{W}\mathbf{W}^T + \sigma^2\mathbf{I}_d$. As shown in (Tipping and Bishop 1999), the maximum likelihood solution for $\mathbf{t}$ just is the mean of observations, and the solution for $\mathbf{W}$ has strong connections to the principal component vectors in conventional PCA. In fact, when $\sigma^2 \to 0$ this probabilistic model recovers PCA.

By introducing a prior distribution over the parameters, a Bayesian PCA model was proposed in (Bishop 1999a), where the effective dimension of latent principal component space can be determined as part of Bayesian inference.

The framework of BM^{2}PCA

From the description above, we can see that the low-dimensional latent representations of data are learned only based on the data covariance. By contrast, here we aim to improve unsupervised PCA learning by exploiting the response values associated with data observations.

First, as in Bayesian PCA (Bishop 1999a), we introduce a prior distribution over the parameters of PPCA and define the following generative process for the n-th observation:

$$\begin{aligned}
\mathbf{t} &\sim \mathcal{N}(\mathbf{t}|\mathbf{0}, \delta^{-1}\mathbf{I}_d)\\
\mathbf{r} &\sim \textstyle\prod_{i=1}^k \Gamma(r_i|a_r, b_r)\\
\mathbf{W} &\sim \textstyle\prod_{i=1}^k \mathcal{N}(\mathbf{w}_i|\mathbf{0}, r_i^{-1}\mathbf{I}_d)\\
\tau &\sim \Gamma(\tau|a_\tau, b_\tau)\\
\mathbf{z}_n &\sim \mathcal{N}(\mathbf{z}_n|\mathbf{0},\ \mathbf{I}_k)\\
\mathbf{x}_n &\sim \mathcal{N}(\mathbf{x}_n|\mathbf{W}\mathbf{z}_n+\mathbf{t},\ \tau^{-1}\mathbf{I}_d)
\end{aligned}$$

where $\Gamma(\cdot)$ is the Gamma distribution[1], and δ, a_r, b_r, a_τ, b_τ are the hyper-parameters. Note that the hierarchical prior on $\mathbf{W}$ and $\mathbf{r}$ is motivated by automatic relevance determination (ARD) (Neal 1995; Tipping 2001), which can control the effective number of retained principal components. Let $\Omega = (\mathbf{t}, \mathbf{W}, \mathbf{r}, \tau, \mathbf{Z})$ denote all the parameters and latent variables, and $p_0(\Omega) = p_0(\mathbf{t})p_0(\mathbf{W}, \mathbf{r})p_0(\tau)p_0(\mathbf{Z})$ be the prior on them. Then we can see that the Bayesian posterior distribution $p(\Omega|\mathbf{X}) = p_0(\Omega)p(\mathbf{X}|\Omega)/p(\mathbf{X})$ can be equivalently obtained by solving the following information theoretical optimization problem:

$$\min_{q(\Omega)\in\mathcal{P}} \mathrm{KL}(q(\Omega)\|p(\Omega|\mathbf{X})) \tag{1}$$

where $\mathrm{KL}(q\|p)$ is the Kullback-Leibler (KL) divergence, and $\mathcal{P}$ is the space of probability distributions. Expanding (1) and ignoring the term unrelated to $q(\Omega)$, we further get

$$\min_{q(\Omega)\in\mathcal{P}} \mathrm{KL}(q(\Omega)\|p_0(\Omega)) - \mathbb{E}_{q(\Omega)}[\mathrm{log}p(\mathbf{X}|\Omega)].$$

Now consider exploiting the response values $\mathbf{y}$ associated with data observations $\mathbf{X}$. In general, we prefer latent representations $\mathbf{Z}$ that on one hand explain the observed data $\mathbf{X}$ well and on the other hand allow us to learn a predictive model, which predicts $\mathbf{y}$ and the responses of new observations as accurate as possible. As well known, maximum margin learning machines such as SVM have arguably good generalization performance. However, their quadratic optimization based formulations make it not trivial to combine them with Bayesian modeling. In this paper, we adopt the posterior regularization (Jaakkola, Meila, and Jebara 1999; Zhu, Ahmed, and Xing 2012; Zhu et al. 2014; Zhu, Chen, and Xing 2014) strategy to incorporate the max-margin principle into the above unsupervised Bayesian PCA model. As a direct way to impose constraints and incorporate knowledge in Bayesian models, posterior regularization is more natural and general than specially designed priors. Now let Θ be the parameter of a max-margin prediction model $\mathcal{M}$, and $q(\Omega, \Theta)$ denote the joint post-data distribution[2] of Ω and Θ. We define the following expected margin loss of $\mathcal{M}$:

$$\mathcal{R}(q(\Omega, \Theta)) = \mathbb{E}_{q(\Omega,\Theta)} l(\mathbf{Z}, \Theta)$$

[1] Throughout this paper, we use its shape-rate parameterization, i.e., $a.$ and $b.$ are the shape and rate parameter respectively.

[2] We use post-data to distinguish it from posterior.

where $l(\mathbf{Z},\Theta)$ is the margin loss of $\mathcal{M}$ on training data $(\mathbf{X},\mathbf{y})$, then our BM^{2}PCA framework can be formulated as

$$\begin{aligned}\min_{q(\Omega,\Theta)\in\mathcal{P}} & \mathrm{KL}(q(\Omega,\Theta)\|p_0(\Omega,\Theta)) - \mathbb{E}_{q(\Omega)}[\log p(\mathbf{X}|\Omega)] \\ & +2C\cdot\mathcal{R}(q(\Omega,\Theta)) \end{aligned} \qquad (2)$$

where $p_0(\Omega,\Theta) = p_0(\Omega)p_0(\Theta)$ is the prior, C is the regularization parameter, the constant 2 is just for convenience, and the expected margin loss $\mathcal{R}(q(\Omega,\Theta))$ has different forms for different learning tasks.

So far, we have developed our max-margin PCA framework with Bayesian posterior regularization. By defining $l(\mathbf{Z},\Theta)$ with hinge loss and ϵ-insensitive loss respectively, this general framework can handle both classification and regression problems. In the following, we consider binary classification problem to exemplify our framework.

Model for classification

Suppose we have a $1\times N$ label vector $\mathbf{y}$, with its element $y_n\in\{+1,-1\}$, $n=1,...,N$. Our goal is to find the post-data distribution $q(\Omega,\Theta)$ under the framework in (2). First we have to define the margin loss for classification. Specifically, as in SVM we want the two classes of data to be separated from each other by a large margin, which gives us a max-margin classification problem in the latent principal components space. Henceforth we define $\tilde{\mathbf{z}} = [\mathbf{z}^T,\ 1]^T$ as the augmented latent representation of observation $\mathbf{x}$, and let $f(\mathbf{x};\tilde{\mathbf{z}},\boldsymbol{\eta}) = \boldsymbol{\eta}^T\tilde{\mathbf{z}}$ be a discriminant function parameterized by $\boldsymbol{\eta}$. We assume the prior of $\boldsymbol{\eta}$ takes the following form

$$\begin{aligned} p(\boldsymbol{\eta}|\nu) &= \mathcal{N}(\boldsymbol{\eta}|\mathbf{0},\nu^{-1}\mathbf{I}_{(k+1)}) \\ \nu\sim p_0(\nu) &= \Gamma(\nu|a_\nu,b_\nu) \end{aligned}$$

where a_ν and b_ν are hyper-parameters and ν plays a similar role as the penalty parameter in SVM. Thus for classification we have $\Theta=(\boldsymbol{\eta},\nu)$ and $p_0(\Theta)=p_0(\boldsymbol{\eta},\nu)=p(\boldsymbol{\eta}|\nu)p_0(\nu)$.

Now for fixed values of $\mathbf{Z}$ and $\boldsymbol{\eta}$, we can compute the margin loss on training data $(\mathbf{X},\mathbf{y})$ by

$$l(\mathbf{Z},\Theta) = \sum_{n=1}^{N}\max(0, 1-y_n f(\mathbf{x}_n)).$$

Since $\mathbf{Z}$ and $\boldsymbol{\eta}$ actually are random variables, we have to average the loss over their joint distribution, i.e., we have the following expected margin loss[3] for classification:

$$\mathcal{R}_c(q(\Omega,\Theta)) = \sum_{n=1}^{N}\mathbb{E}_{q(\Omega,\Theta)}\max(0,1-y_n\boldsymbol{\eta}^T\tilde{\mathbf{z}}_n).$$

Directly solving (2) with $\mathcal{R}_c$ is difficult and inefficient. Here we regard

$$\varphi(y_n|\tilde{\mathbf{z}}_n,\boldsymbol{\eta}) = \exp\{-2C\cdot\max(0,1-y_n\boldsymbol{\eta}^T\tilde{\mathbf{z}}_n)\}$$

as the unnormalized pseudo-likelihood of the label variable for the n-th data, then our model can be rewritten as

$$\begin{aligned}\min_{q(\Omega,\Theta)\in\mathcal{P}} & \mathrm{KL}(q(\Omega,\Theta)\|p_0(\Omega,\Theta)) - \mathbb{E}_{q(\Omega)}[\log p(\mathbf{X}|\Omega)] \\ & -\mathbb{E}_{q(\Omega,\Theta)}[\log(\varphi(\mathbf{y}|\mathbf{Z},\boldsymbol{\eta}))] \end{aligned} \qquad (3)$$

where $\varphi(\mathbf{y}|\mathbf{Z},\boldsymbol{\eta}) = \prod_{n=1}^{N}\varphi(y_n|\tilde{\mathbf{z}}_n,\boldsymbol{\eta})$. Solving problem (3), we can get the posterior distribution

$$q(\Omega,\Theta) = \frac{p_0(\Omega,\Theta)p(\mathbf{X}|\Omega)\varphi(\mathbf{y}|\mathbf{Z},\boldsymbol{\eta})}{\phi(\mathbf{X},\mathbf{y})},$$

where $\phi(\mathbf{X},\mathbf{y})$ is the normalization constant, which is intractable to compute analytically due to the max function in φ. In the following, we develop an efficient data augmentation based variational algorithm to approximate $q(\Omega,\Theta)$.

Variational approximate inference

Since directly solving for the posterior is intractable, we appeal to the variational approximate Bayesian inference method (Beal 2003; Bishop 2006) which is generally much more efficient than the Markov Chain Monte Calo (MCMC) based sampling methods, and thus allows us to scale to large data sets.

First, to deal with the max function in $\varphi(\cdot)$, we apply the data augmentation idea (Polson and Scott 2011; Tanner and Wong 1987) and transform the pseudo-likelihood function into the integration of a function with augmented variable:

$$\varphi(y_n|\tilde{\mathbf{z}}_n,\boldsymbol{\eta}) = \int_0^\infty \frac{\exp\{\frac{-1}{2\lambda_n}[\lambda_n + C(1-y_n\boldsymbol{\eta}^T\tilde{\mathbf{z}}_n)]^2\}}{\sqrt{2\pi\lambda_n}}d\lambda_n.$$

Let

$$\varphi(\mathbf{y},\boldsymbol{\lambda}|\mathbf{Z},\boldsymbol{\eta}) = \prod_{n=1}^{N}\frac{\exp\{\frac{-1}{2\lambda_n}[\lambda_n + C(1-y_n\boldsymbol{\eta}^T\tilde{\mathbf{z}}_n)]^2\}}{\sqrt{2\pi\lambda_n}},$$

then we can get the augmented posterior distribution[4]

$$q(\Omega,\Theta,\boldsymbol{\lambda}) \propto p_0(\Omega,\Theta)p(\mathbf{X}|\Omega)\varphi(\mathbf{y},\boldsymbol{\lambda}|\mathbf{Z},\boldsymbol{\eta}). \qquad (4)$$

In the following we will approximate this augmented posterior with the mean-field variational method. Specifically, we assume there are a family of fully factorized but free-form variational distributions

$$V(\Omega,\Theta,\boldsymbol{\lambda}) = V(\mathbf{t})V(\mathbf{W})V(\mathbf{r})V(\tau)V(\mathbf{Z})V(\boldsymbol{\eta})V(\boldsymbol{\lambda})V(\nu)$$

and the goal is to get the optimal one which minimizes the KL divergence $\mathrm{KL}(V(\Omega,\Theta,\boldsymbol{\lambda})\|q(\Omega,\Theta,\boldsymbol{\lambda}))$ between the approximating distribution and the target posterior. To achieve this, our strategy is to first initialize the moments of all factor distributions of $V(\Omega,\Theta,\boldsymbol{\lambda})$ appropriately and then iteratively optimize each of the factors in turn using the current estimates for all of the other factors. Convergence is guaranteed because the KL divergence is convex with respect to each of the factors. Now let us first expand the right side of (4) and get the joint distribution of data and parameters as follows

$$\begin{aligned} p(\Omega,\Theta,\boldsymbol{\lambda},\mathbf{X},\mathbf{y}) &= p_0(\mathbf{t})p(\mathbf{W}|\mathbf{r})p_0(\mathbf{r})p_0(\tau)p_0(\mathbf{Z})p(\boldsymbol{\eta}|\nu) \\ &\quad \cdot p_0(\nu)p(\mathbf{X}|\mathbf{t},\mathbf{W},\tau,\mathbf{Z})\varphi(\mathbf{y},\boldsymbol{\lambda}|\mathbf{Z},\boldsymbol{\eta}). \end{aligned}$$

Then it can be shown that when keeping all other factors fixed the optimal distribution $V^*(\mathbf{Z})$ satisfies

$$V^*(\mathbf{Z}) \propto \exp\{\mathbb{E}_{-\mathbf{Z}}[\log p(\Omega,\Theta,\boldsymbol{\lambda},\mathbf{X},\mathbf{y})]\} \qquad (5)$$

[3] Expected margin loss (Zhu et al. 2014) is more convenient and upper-bounds the margin loss of the expected prediction model (Jaakkola, Meila, and Jebara 1999) by Jensen's inequality.

[4] Its conditionals have convenient forms and its marginalization over $\boldsymbol{\lambda}$ recovers $q(\Omega,\Theta)$.

where $\mathbb{E}_{-\mathbf{Z}}$ denotes the expectation with respect to $V(\Omega, \Theta, \boldsymbol{\lambda})$ over all variables except for $\mathbf{Z}$. Plugging all involved quantities into (5), we can further get

$$
\begin{aligned}
V^*(\mathbf{Z}) &= \prod_{n=1}^{N} \mathcal{N}(\mathbf{z_n}|\mu_{\mathbf{z}}^{(n)}, \Sigma_{\mathbf{z}}^{(n)}) \\
\Sigma_{\mathbf{z}}^{(n)} &= \{C^2 \mathbb{E}_{\boldsymbol{\eta}}[\tilde{\boldsymbol{\eta}}\tilde{\boldsymbol{\eta}}^T]\mathbb{E}_{\boldsymbol{\lambda}}[\lambda_n^{-1}] + \mathbf{I}_k \\
&\quad + \mathbb{E}_\tau[\tau]\mathbb{E}_{\mathbf{W}}[\mathbf{W^T W}]\}^{-1} \\
\mu_{\mathbf{z}}^{(n)} &= \Sigma_{\mathbf{z}}^{(n)}\{\mathbb{E}_\tau[\tau]\mathbb{E}_{\mathbf{W}}[\mathbf{W}^T](\mathbf{x}_n - \mathbb{E}_{\mathbf{t}}[\mathbf{t}]) \\
&\quad + \mathbb{E}_{\boldsymbol{\lambda}}[\lambda_n^{-1}]\{C(\mathbb{E}_{\boldsymbol{\lambda}}[\lambda_n] + C)y_n \mathbb{E}_{\boldsymbol{\eta}}[\tilde{\boldsymbol{\eta}}] \\
&\quad - C^2 \mathbb{E}_{\boldsymbol{\eta}}[\eta_{(k+1)}\tilde{\boldsymbol{\eta}}]\}\}
\end{aligned}
$$

where $\tilde{\boldsymbol{\eta}}$ denotes the first k dimensions of $\boldsymbol{\eta}$, i.e., $\boldsymbol{\eta} = [\tilde{\boldsymbol{\eta}},\ \eta_{(k+1)}]$. Similarly, we can get the updating equations for all other factors. Since they are tedious and easy to derive, here we only provide the equations for $\boldsymbol{\lambda}$, ν, and $\boldsymbol{\eta}$:

$$
\begin{aligned}
V^*(\boldsymbol{\lambda}) &= \prod_{n=1}^{N} \mathcal{GIG}(\lambda_n|\frac{1}{2}, 1, \chi^{(n)}) \\
\chi^{(n)} &= C^2(1 - y_n \mathbb{E}_{\boldsymbol{\eta}}[\boldsymbol{\eta}^T]\mathbb{E}_{\mathbf{Z}}[\tilde{\mathbf{z}}_n])^2 \\
V^*(\nu) &= \Gamma(\nu|\tilde{a}_\nu, \tilde{b}_\nu) \\
\tilde{a}_\nu &= a_\nu + L/2, \\
\tilde{b}_\nu &= b_\nu + \mathbb{E}_{\boldsymbol{\eta}}[\|\boldsymbol{\eta}\|^2]/2 \\
V^*(\boldsymbol{\eta}) &= \mathcal{N}(\boldsymbol{\eta}|\mu_{\boldsymbol{\eta}}, \Sigma_{\boldsymbol{\eta}}) \\
\Sigma_{\boldsymbol{\eta}} &= \{C^2 \sum_{n=1}^{N} \mathbb{E}_{\mathbf{Z}}[\tilde{\mathbf{z}}_n \tilde{\mathbf{z}}_n^T]\mathbb{E}_{\boldsymbol{\lambda}}[\lambda_n^{-1}] \\
&\quad + \mathbb{E}_\nu[\nu]\mathbf{I}_{(k+1)}\}^{-1} \\
\mu_{\boldsymbol{\eta}} &= \Sigma_{\boldsymbol{\eta}} \sum_{n=1}^{N} C(1 + C\mathbb{E}_{\boldsymbol{\lambda}}[\lambda_n^{-1}])y_n \mathbb{E}_{\mathbf{Z}}[\tilde{\mathbf{z}}_n]
\end{aligned}
$$

where $\mathcal{GIG}(\cdot)$ is the generalized inverse Gaussian distribution. The equations for $\mathbf{t}$, $\mathbf{W}$, $\mathbf{r}$ and τ are similar as those in (Bishop 1999b), thus are omitted here.

Prediction on unseen data

Suppose we have a set of test data that is unseen during the model training phase. The goal is to predict the labels of these data as accurate as possible. To apply our classification model learned above, we have to first project the new data to the same low-dimensional feature space as that for training data. Given the optimal variational distributions $V^*(\mathbf{t})$, $V^*(\mathbf{W})$, and $V^*(\tau)$ learned in the training phase, we use a single step variational method to approximate the posterior latent representation $p(\mathbf{z}_{new}|\mathbf{x}_{new})$ for test data $\mathbf{x}_{new}$:

$$
\begin{aligned}
V^*(\mathbf{z}_{new}) &= \mathcal{N}(\mathbf{z}_{new}|\mu_{\mathbf{z}}^{new}, \Sigma_{\mathbf{z}}^{new}) \\
\Sigma_{\mathbf{z}}^{new} &= \{\mathbf{I}_k + \mathbb{E}_\tau[\tau]\mathbb{E}_{\mathbf{W}}[\mathbf{W^T W}]\}^{-1} \\
\mu_{\mathbf{z}}^{new} &= \Sigma_{\mathbf{z}}^{new}\mathbb{E}_\tau[\tau]\mathbb{E}_{\mathbf{W}}[\mathbf{W}^T](\mathbf{x}_{new} - \mathbb{E}_{\mathbf{t}}[\mathbf{t}])
\end{aligned}
$$

where the expectations are taken over the optimal variational distributions of $\mathbf{t}$, $\mathbf{W}$, and τ.

Then with the optimal variational approximation $V^*(\boldsymbol{\eta})$ for the posterior distribution of classification parameter $\boldsymbol{\eta}$, we can predict the class label of $\mathbf{x}_{new}$ by

$$
\begin{aligned}
\tilde{\mu}_{\mathbf{z}}^{new} &= [(\mu_{\mathbf{z}}^{new})^T,\ 1]^T \\
y_{new} &= \mathrm{sgn}(\mathbb{E}_{\boldsymbol{\eta}, \mathbf{z}_{\mathrm{new}}}[\boldsymbol{\eta}^T \tilde{\mathbf{z}}_{new}]) \\
&= \mathrm{sgn}(\mu_{\boldsymbol{\eta}}^T \tilde{\mu}_{\mathbf{z}}^{new}).
\end{aligned}
$$

Computational complexity

For each iteration of the variational inference on training data, we need $\mathrm{O}(Ndk^4)$ computation, most of which is spent on the calculation of $\Sigma_{\mathbf{z}}^{(n)},\ n = 1, ..., N$ where the inversion of each covariance matrix consumes $\mathrm{O}(k^3)$ computation. However, noting that in typical uses, k usually is very small, e.g., 10 or 20, our model can be approximatively seen as scaling linearly in the training size N and original dimensionality d. For testing on unseen data with N_{test} test samples, we only need to invert the covariance matrix $\Sigma_{\mathbf{z}}$ one time, so the complexity is $\mathrm{O}(N_{test} + k^3)dk$.

Experiments

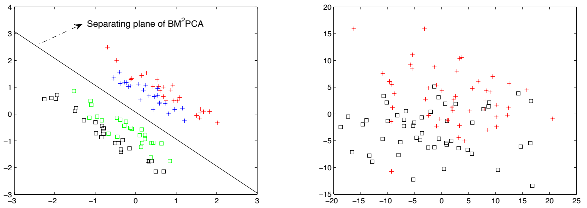
We evaluate the proposed BM^{2}PCA model on various classification tasks. Note that for real tasks, the classification problems typically have multiple classes, so though our model is designed for binary classification, we adapted it with the one-VS-rest strategy like that for SVM.

Parameter setting

In all of our experiments, the hyper-parameters of BM^{2}PCA are set as: $a_r = b_r =$ 1e-3, $a_\tau =$ 1e-2, $a_\nu =$ 1e-1, $b_\tau = b_\nu = \delta =$ 1e-5. For the regularization parameter C, we empirically found that BM^{2}PCA works well on most of our data sets when $10 \leq C \leq 40$. We decide to select C from the integer set $\{10, 20, 30, 40\}$ for each data set by performing L-fold cross-validation on training data, where L is the smaller one of 5 and the number of training samples per class.

Illustration on synthetic data

We generate two Gaussian clusters of 50 data points in a 2-dimensional space, with each corresponding to one class. Then we add three other dimensions to each point by sampling from a given multivariate Gaussian. For PCA, we use all the 100 points to learn a projection in 2-dimension space, while a random and equal split into training/testing is conducted for BM^{2}PCA. As shown in Figure 1, the points in red and black are training samples, and the points in blue and green are testing ones. We use crosses and squares to indicate positive and negative samples respectively. It is easy to see that BM^{2}PCA found a good subspace for both training and testing while PCA worked not so well.



(a) BM^{2}PCA (b) PCA

Figure 1: Projection results on synthetic data: (a) BM^{2}PCA; (b) PCA. Crosses and squares are positive and negative samples respectively. Points in red and black are training samples while points in blue and green are testing ones.

Real Data sets

We test BM^{2}PCA on video retrieval, face recognition, gene classification and text categorization problems. Some statistics of these data sets are shown in Table 1. For the TRECVID2003 data, we have 1078 manually labeled video shots each of which is represented by a 1894-dimension binary vector of text features and a 165-dimension vector of HSV color histogram. The Yale data contains 165 gray scale face images in GIF format of 15 individuals. There are 11 images per individual, one per different facial expression or configuration such as center-light, left-light, happy or surprised. The ORL data contains 10 different face images for each of 40 distinct subjects. For some subjects, the images were taken at different times with varying lighting and facial details. The YaleB (the extended Yale Face Database B) data includes 38 individuals and about 64 near frontal face images under different illuminations per individual. All faces are manually aligned, cropped and resized to 32×32 or 64×64 pixels. For the 11 Tumors and 14 Tumors gene expression data sets, we have 11 various human tumor types, and 14 various human tumor types with 12 normal tissue types respectively. The characteristics of these data are their high dimensionality and small samples. Finally, the 20 Newsgroups data contains 20,000 news articles posted in 20 newsgroups. After removing the words that occur less than 5 times, we have 19,928 documents with 25,284 words.

Table 1: Statistics of the multi-class data sets.

	Category	#Data	#Dim	#Class
TRECVID2003	Video	1078	2059	5
Yale	Face	165	4096	15
ORL	Face	400	1024	40
YaleB	Face	2414	1024	38
11 Tumors	Gene	174	12533	11
14 Tumors	Gene	308	15009	26
20 Newsgroups	Text	19928	25284	20

Evaluation and results

Competitors (1) Six state-of-the-art supervised dimensionality reduction methods: supervised probabilistic PCA (SPPCA) (Yu et al. 2006), supervised exponential family PCA (SEPCA) (Guo 2009), supervised dimensionality reduction with generalized linear models (SDR-GLM) (Rish et al. 2008), Maximum margin supervised topic models (MedLDA) (Zhu, Ahmed, and Xing 2012), large-margin Harmonium (MMH) (Chen et al. 2012), and infinite latent SVM (iLSVM) (Zhu, Chen, and Xing 2014); and (2) three baseline methods: direct SVM learning in original feature space (FULL), SVM learning in principal component space (PCA), and SVM learning in the space given by linear discriminant analysis (LDA). For multiclass SVM (Crammer and Singer 2002), we use a fast implementation from the LIBLINEAR[5] package (Fan et al. 2008).

Evaluation To compare with SPPCA, we conduct experiments on the ORL, 14 Tumors, and 20 Newsgroups data sets that are used in its original paper (Yu et al. 2006). Our data organization is the same as theirs, i.e., each sample is normalized to have unit length, and TF-IDF features are used for 20 Newsgroups data. The number of training samples per class is 2 for ORL and 14 Tumors, and 5 for 20 Newsgroups. For all projection methods, the data are projected into 10-dimensional space. The results of BM^{2}PCA, FULL and PCA are averaged over 20 independent runs and shown in Table 2. Here we also provide the result of MedLDA, a latest supervised topic model for text with maximum margin principle, but note that it can only address word count data. From the results we can see BM^{2}PCA has obvious advantage over all competitors on ORL and 14 Tumors data, and only performs a little worse than the decoupled PCA and SVM learning on 20 Newsgroups. However, it should be noted that PCA used both labeled and unlabeled data to learn the projection, while BM^{2}PCA and other SDR methods only used labeled ones. Considering 20 Newsgroups is a very large data set and thus the few training samples cannot reflect it well, the performance of BM^{2}PCA is non-trivial.

Table 2: Comparison on multi-class data sets with unit length normalization. Listed results are test accuracies (%) averaged over 20 independent runs. Bold face indicates highest accuracy.

	ORL	14 Tumors	20 News	Average
FULL	41.7 ± 8.7	53.4 ± 2.5	45.3 ± 1.4	46.8
PCA	54.4 ± 2.9	34.5 ± 3.4	**38.8 ± 2.5**	42.5
LDA	19.1 ± 2.5	34.7 ± 4.8	31.1 ± 2.6	28.3
SPPCA	61.7 ± 4.1	36.8 ± 3.6	10.7 ± 2.4	36.4
MedLDA	-	-	14.2 ± 2.8	14.2
BM^{2}PCA	**73.7 ± 3.8**	**54.3** ± **3.8**	35.3 ± 3.0	**54.4**

Different from SPPCA, the convex SEPCA model proposed in (Guo 2009) assumes each feature of the data is centered to have zero mean. Here we also give comparisons with it on the Yale, YaleB and 11 Tumors data, where the number of training samples per class is 3 for Yale and 11 Tumors, and 5 for YaleB. Again, for all projection methods, the data are projected into 10-dimensional space. The averaged results are shown in Table 3, from which we can see BM^{2}PCA almost always outperforms other SDR competitors and the decoupled PCA and SVM learning method. SEPCA achieved excellent performance on the 11 Tumors

[5] Available at: http://www.csie.ntu.edu.tw/%7Ecjlin/liblinear/.

data, which may due to its convex formulation and the global optimum, however, its performance on the YaleB data is not so good because of its maximum likelihood principle. By contrast, BM^{2}PCA always is among the best, yielding highest overall accuracy on three data sets.

Table 3: Comparison on centered data sets with test accuracies (%) averaged over 20 independent runs. The results for SEPCA, SDR-GLM and SPPCA are cited from (Guo 2009).

	Yale	YaleB	11 Tumors	Average
FULL	74.2 ± 3.1	62.3 ± 6.8	83.8 ± 3.7	73.4
PCA	55.8 ± 4.2	12.9 ± 5.3	67.6 ± 6.3	45.4
LDA	37.1 ± 7.1	15.7 ± 1.8	28.6 ± 5.2	27.1
SPPCA	51.6	9.8	63.0	41.5
SDR-GLM	58.8	19.0	63.5	47.1
SEPCA	64.4	20.5	**88.9**	57.9
BM^{2}PCA	**65.7 ± 3.5**	**43.8 ± 4.8**	77.1 ± 4.9	**62.2**

We also compare BM^{2}PCA with the infinite latent SVM (iLSVM) (Zhu, Chen, and Xing 2014), large-margin Harmonium (MMH) (Chen et al. 2012) and a decoupled approach of EFH+SVM on the TRECVID2003 data set. EFH+SVM uses the exponential family Harmonium (EFH) (Welling, Rosen-Zvi, and Hinton 2004) to discover latent features and then learns a multiclass SVM. Here we use the same training/testing split as in (Chen et al. 2012), and like in (Zhu, Chen, and Xing 2014) we only consider the real-valued HSV features. We set the number of components to $k = 10$ for BM^{2}PCA, and the results in terms of accuracy and F1 score are shown in Table 4, from which we can see BM^{2}PCA achieves the best performance.

Table 4: Results (%) on TRECVID2003 data. BM^{2}PCA, MMH and EFH have zero std due to their deterministic initialization.

	EFH+SVM	MMH	iLSVM	BM^{2}PCA
Accuracy	56.5 ± 0.0	56.6 ± 0.0	56.3 ± 1.0	**63.8 ± 0.0**
F1 score	42.7 ± 0.0	43.0 ± 0.0	44.8 ± 1.1	**47.6 ± 0.0**

Sensitivity analysis

We study the sensitivity of BM^{2}PCA with respect to sampling ratio, component number k, and the parameter C.

Sampling ratio First, we show the performance improvement of BM^{2}PCA with increasing number of training samples. Here we take the Yale data as example and fix C to be 10. As the averaged results over 20 runs show in Figure 2, BM^{2}PCA (with different number of principal components) performs better when more training samples are available, which is the desired property for most applications. For comparison, we also provide the results of SVM learning in original feature space, which are consistently worse than those of BM^{2}PCA with $k = 30$.

Number of components Also from Figure 2, we can observe that the performance of BM^{2}PCA increase steadily when more principal components are learned (similar trends are shown in Figure 3 with different parameter C). The results of decoupled PCA and SVM learning are given in Figure 2 as well. We can find that BM^{2}PCA outperforms the decoupled method significantly, no matter how many components and training samples are used.

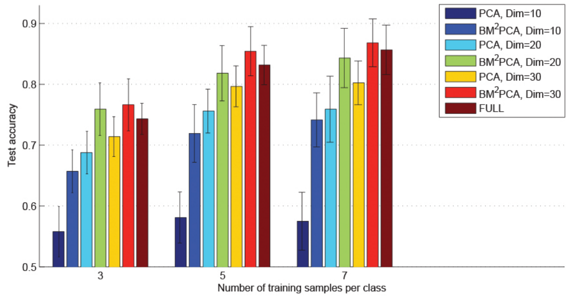
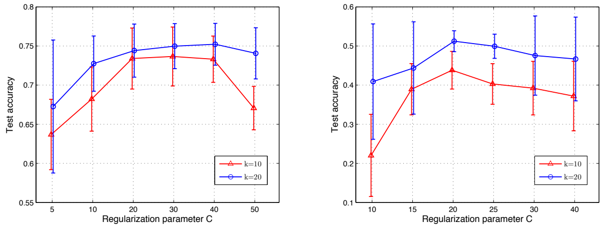


Figure 2: Results on Yale data set with different sampling ratio and number of components k (dimensions).

Regularization parameter C Finally, we show how the regularization parameter C influences the prediction performance of BM^{2}PCA. We use 2 and 5 training samples per class for the ORL and YaleB data respectively. The averaged results over 20 runs of BM^{2}PCA are shown in Figure 3, where we considered different number of components, i.e., $k = 10$ and $k = 20$. We can see that while different data sets prefer different C, different k seem have similar interests of C for a given data set.



(a) ORL data

(b) YaleB data

Figure 3: Effect of regularization parameter C of BM^{2}PCA with different k: (a) ORL data; (b) YaleB data.

Conclusions and future work

We presented a Bayesian approach to combine PCA with max-margin learning. Under the Bayesian framework, our method can infer the weight and penalty parameter of max-margin machine while finding the most appropriate principal components simultaneously. Experiments on various classification tasks show the superiority of our method.

Our framework can be extended in several aspects. First, it is natural to conduct semi-supervised learning by extracting principal components on all observed samples while training classification model only on those labeled ones. Second, we can also define the expected margin loss $\mathcal{R}(q(\Omega, \Theta))$ for regression problem similar as in ϵ-insensitive Support Vector Regression, and our data augmentation based variational inference can be easily adapted to this case. Third, it is also interesting to extend BM^{2}PCA to deal with multi-view data (Chen et al. 2012) and multi-task data (Evgeniou and Pontil 2007). These will be the promising future work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61473273, 61473274, 61175052, 61203297), National High-tech R&D Program of China (863 Program) (No.2014AA015105, 2013AA01A606, 2012AA011003). Changying Du was also supported by the China Scholarship Council for one year study at Purdue University, West Lafayette, USA. We thank the anonymous reviewers and prof. Dongbo Bu from Institute of Computing Technology, CAS for their helpful comments.

References

Beal, M. J. 2003. *Variational algorithms for approximate Bayesian inference*. Ph.D. Dissertation, University of London.

Bishop, C. M. 1999a. Bayesian pca. In *Advances in neural information processing systems*, 382–388.

Bishop, C. M. 1999b. Variational principal components. In *Proceedings of the 9th International Conference on Artificial Neural Networks, ICANN99*.

Bishop, C. M. 2006. *Pattern recognition and machine learning*, volume 1. Springer New York.

Chen, N.; Zhu, J.; Sun, F.; and Xing, E. P. 2012. Large-margin predictive latent subspace learning for multiview data analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34(12):2365–2378.

Crammer, K., and Singer, Y. 2002. On the algorithmic implementation of multiclass kernel-based vector machines. *The Journal of Machine Learning Research* 2:265–292.

Evgeniou, A., and Pontil, M. 2007. Multi-task feature learning. *Advances in neural information processing systems* 41–48.

Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; and Lin, C.-J. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research* 9:1871–1874.

Fukumizu, K.; Bach, F. R.; and Jordan, M. I. 2004. Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces. *The Journal of Machine Learning Research* 5:73–99.

Ghahramani, Z., and Griffiths, T. L. 2005. Infinite latent feature models and the indian buffet process. In *Advances in Neural Information Processing Systems*, 475–482.

Guo, Y. 2009. Supervised exponential family principal component analysis via convex optimization. In *Advances in Neural Information Processing Systems*, 569–576.

Jaakkola, T.; Meila, M.; and Jebara, T. 1999. Maximum entropy discrimination. In *Advances in neural information processing systems*.

Mcauliffe, J. D., and Blei, D. M. 2008. Supervised topic models. In *Advances in neural information processing systems*, 121–128.

Neal, R. M. 1995. *Bayesian learning for neural networks*. Ph.D. Dissertation, University of Toronto.

Pereira, F., and Gordon, G. 2006. The support vector decomposition machine. In *Proceedings of the 23rd international conference on Machine learning*, 689–696. ACM.

Polson, N. G., and Scott, S. L. 2011. Data augmentation for support vector machines. *Bayesian Analysis* 6(1):1–23.

Raeder, T.; Perlich, C.; Dalessandro, B.; Stitelman, O.; and Provost, F. 2013. Scalable supervised dimensionality reduction using clustering. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1213–1221. ACM.

Rish, I.; Grabarnik, G.; Cecchi, G.; Pereira, F.; and Gordon, G. J. 2008. Closed-form supervised dimensionality reduction with generalized linear models. In *Proceedings of the 25th international conference on Machine learning*, 832–839. ACM.

Tanner, M. A., and Wong, W. H. 1987. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association* 82(398):528–540.

Tipping, M. E., and Bishop, C. M. 1999. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 61(3):611–622.

Tipping, M. E. 2001. Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research* 1:211–244.

Welling, M.; Rosen-Zvi, M.; and Hinton, G. E. 2004. Exponential family harmoniums with an application to information retrieval. In *Advances in neural information processing systems*, 1481–1488.

Xu, C.; Tao, D.; Xu, C.; and Rui, Y. 2014. Large-margin weakly supervised dimensionality reduction. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 865–873.

Yu, S.; Yu, K.; Tresp, V.; Kriegel, H.-P.; and Wu, M. 2006. Supervised probabilistic principal component analysis. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 464–473. ACM.

Zhang, D.; Zhou, Z.-H.; and Chen, S. 2007. Semi-supervised dimensionality reduction. In *SDM*, 629–634. SIAM.

Zhu, J.; Ahmed, A.; and Xing, E. P. 2012. Medlda: maximum margin supervised topic models. *Journal of Machine Learning Research* 13(1):2237–2278.

Zhu, J.; Chen, N.; Perkins, H.; and Zhang, B. 2014. Gibbs max-margin topic models with data augmentation. *Journal of Machine Learning Research* 15:1073–1110.

Zhu, J.; Chen, N.; and Xing, E. P. 2014. Bayesian inference with posterior regularization and applications to infinite latent svms. *Journal of Machine Learning Research* 15:1799–1847.