Supervised Exponential Family Principal Component Analysis via Global Optimization

Yuhong Guo
yuhong@temple.edu
Computer and Information Sciences
Temple University
Philadelphia, PA 19122, USA

Abstract

Recently, supervised dimensionality reduction has been gaining attention, owing to the realization that data labels are often available and indicate important underlying structure in the data. In this paper, we present a novel convex supervised dimensionality reduction approach based on exponential family PCA, which is able to avoid the local optima of typical EM learning. Moreover, by introducing a sample-based approximation to exponential family models, it overcomes the limitation of the prevailing Gaussian assumptions of standard PCA, and produces a kernelized formulation for nonlinear supervised dimensionality reduction. A training algorithm is then devised based on a subgradient bundle method, whose scalability can be gained using a coordinate descent procedure. The advantage of our global optimization approach is demonstrated by empirical results over both synthetic and real data. [1]

1 Introduction

Principal component analysis (PCA) has been extensively used for data analysis and processing. It provides a closed-form solution for linear unsupervised dimensionality reduction through singular value decomposition (SVD) on the data matrix [7]. Probabilistic interpretations of PCA have also been provided in [8, 18], which formulate PCA using a latent variable model with Gaussian distributions. To generalize PCA to better suit non-Gaussian data, many extensions to PCA have been proposed that relax the assumption of a Gaussian data distribution. Exponential family PCA is the most prominent example, where the underlying dimensionality reduction principle of PCA is extended to the general exponential family [4, 6, 15]. Previous work has shown that improved quality of dimensionality reduction can be obtained by using exponential family models appropriate for the data at hand [4, 15]. Given data from a non-Gaussian distribution these techniques are better able than PCA to capture the intrinsic low dimensional structure. However, most existing

[1] A preliminary version of this paper appears in NIPS08.

non-Gaussian dimensionality reduction methods rely on iterative local optimization procedures and thus suffer from local optima, with the sole exception of [6] which shows a general convex form can be obtained for dimensionality reduction with exponential family models.

Recently, supervised dimensionality reduction has begun to receive increased attention. As the goal of dimensionality reduction is to identify the intrinsic structure of a data set in a low dimensional space, there are many reasons why supervised dimensionality reduction is a meaningful topic to study. First, data labels are almost always assigned based on some important intrinsic property of the data. Such information should be helpful to suppress noise and capture the most useful aspects of a compact representation of the data. Moreover, there are many high dimensional data sets with label information available, e.g., face and digit images, and it is unwise to ignore them. A few supervised dimensionality reduction methods based on exponential family models have been proposed in the literature. For example, a supervised probabilistic PCA (SPPCA) model was proposed in [21]. SPPCA extends probabilistic PCA by assuming that both features and labels have Gaussian distributions and are generated independently from the latent low dimensional space through linear transformations. The model is learned by maximizing the marginal likelihood of the observed data using an alternating EM procedure. A more general supervised dimensionality reduction approach with generalized linear models (SDR_GLM) was proposed in [13]. SDR_GLM views both features and labels as exponential family random variables and optimizes a weighted linear combination of their conditional likelihood given latent low dimensional variables using an alternating EM-style procedure with closed-form update rules. SDR_GLM is able to deal with different data types by using different exponential family models. Similar to SDR_GLM, the linear supervised dimensionality reduction method proposed in [16] also takes advantage of exponential family models to deal with different data types. However, it optimizes the conditional likelihood of labels given observed features within a mixture model framework using an EM-style optimization procedure. Beyond the PCA framework, many other supervised dimensionality reduction methods have been proposed in the literature. Linear (fisher) discriminant analysis (LDA) is a popular alternative [5], which maximizes between-class variance and minimizes within-class variance. Moreover, a kernelized fisher discriminant analysis (KDA) has been studied in [10]. Another notable nonlinear supervised dimensionality reduction approach is the colored maximum variance unfolding (MVU) approach proposed in [17], which maximizes the variance aligning with the side information (e.g., label information), while preserving the local distance structures from the data. However, colored MVU has only been evaluated on training data.

In this paper, we propose a novel supervised exponential family PCA model (SEPCA). In the SEPCA model, observed data $\mathbf{x}$ and its label y are assumed to be generated from the latent variables $\mathbf{z}$ via conditional exponential family models; dimensionality reduction is conducted by optimizing the conditional likelihood of the observations $(\mathbf{x}, y)$. By exploiting convex duality of the sub-problems and eigenvector properties, a solvable convex formulation of the problem can be derived that forms an upper bound in general, but preserves solution equivalence under an additional assumption, to the original. This convex formulation allows efficient global optimization algorithms to be devised. Moreover, by introducing a sample-based approximation to exponential family models, SEPCA does not suffer from the limitations of implicit Gaussian assumptions and is able to be conveniently kernelized to achieve nonlinearity. A training algorithm is then devised based on a subgradient bundle method, whose scalability can be gained through a coordinate descent procedure. Finally, we present a simple formulation to project new testing data into the embedded space. This projection can be used for other supervised dimensionality reduction approach as well. Our experimental results over both synthetic and real data suggest that a more global, principled probabilistic approach, SEPCA, is better able to capture subtle structure in the data, particularly when good label information is present.

The remainder of this paper is organized as follows. First, in Section 2 we introduce preliminaries about PCA and exponential family PCA models. Then, in Section 3 we present the proposed supervised exponential family PCA model and formulate a convex nondifferentiable optimization problem. An efficient global optimization algorithm is presented in Section 4. In Section 5, we present a simple projection method for new testing points. We then present the experimental results in Section 6. Finally, in Section 7 we conclude the paper.

2 Preliminaries

Given a $t \times n$ data matrix X, consisting of t observations of n-dimensional feature vectors, $X_{i:}$, we consider to obtain its low dimensional representation Z, which is a $t \times d$ matrix for $d < n$. For simplicity, we assume features in X are centered; that is, their empirical means are zeros. The standard PCA finds the low dimensional subspace by minimizing the reconstruction error $\mathrm{tr}((X - ZW)(X - ZW)^\top)$, where W is a $d \times n$ parameter matrix. PCA has a probabilistic interpretation [18], where each point $X_{i:}$ is taken as a random draw from a unit Gaussian with mean $\Theta_{i:}$ for a $t \times n$ matrix Θ, which can be further viewed as a linear mapping of the low dimensional Z such that $\Theta = ZW$.

The Gaussian assumption of the standard PCA may be inappropriate for non-Gaussian data. For example, if data is binary-valued, the Bernoulli distribution may be a better option; if data is nonnegative and integer-valued, the Poisson distribution may be used instead. This drawback of the standard PCA has been noticed by a few researchers. To address this problem, exponential family PCA methods that generalize the standard PCA to the exponential family have been studied [4, 15]. The main idea is to use a unified exponential family representation for a set of distributions that fall into the class of exponential family, thus one can maintain flexibilities on dealing with different types of data. Exponential family [9] denotes a family of distributions that can be written in the form

$$\log P(x|\theta) = \log P_0(x) + x\theta - A(\theta)$$

where θ is called the natural parameter and is usually a real value; $A(\theta)$ is a function that ensures that the probabilities sum or integrate to one over the domain of x, such as $A(\theta) = \log \int_{x \in \mathcal{X}} P_0(x) \exp(x\theta)$. Here $\mathcal{X}$ denotes the domain of x. $P_0(x)$ is a term that depends only on x, and usually can be ignored as a constant during parameter estimation. Gaussian distributions, Binomial distributions and Possion distributions are all members of this family. The main difference between different members of the exponential family is the form of $A(\theta)$ [4].

In this paper, we will in particular consider a general exponential family representation of the conditional distribution of a $n \times 1$ observation vector $\mathbf{x}$ given a $d \times 1$ low dimensional vector $\mathbf{z}$ and a $d \times n$ parameter matrix W:

$$\log P(\mathbf{x}|\mathbf{z}, W) = \log P_0(\mathbf{x}) + \mathbf{x}^\top W^\top \mathbf{z} - A(W^\top \mathbf{z})$$

where

$$A(W^\top \mathbf{z}) = \log \int P_0(\mathbf{x}) \exp(\mathbf{x}^\top W^\top \mathbf{z}) d\mathbf{x}$$

Here the natural parameters $\boldsymbol{\theta}$ can be defined as $\boldsymbol{\theta} = W^\top \mathbf{z}$. When this conditional distribution is a Gaussian distribution with unit variance, we have $\log P_0(\mathbf{x}) = -\frac{1}{2}(n \log(2\pi) + \mathbf{x}^\top \mathbf{x})$, and $A(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^\top \boldsymbol{\theta}$. Nevertheless, we would like to maintain the general exponential family represenation to allow flexibilities in addressing different types of data. Given the observation data matrix X and its low dimensional represenation Z defined before, the conditional likelihood of the data using exponential family represenation can be written as

$$\log P(X|Z, W) = \sum_i \log P_0(X_{i:}) + \operatorname{tr}\left(ZWX^\top\right) - \sum_i A(Z_{i:}W) \tag{1}$$

where

$$A(Z_{i:}W) \quad = \quad \log \int \exp\left(Z_{i:}W\mathbf{x}\right)\ P_0(\mathbf{x})\ d\mathbf{x}\ . \tag{2}$$

3 Supervised Exponential Family PCA

In this section, we consider to extend exponential family PCA to address the problem of supervised dimensionality reduction. Same as above, we assume we are given a $t \times n$ data matrix, X, consisting of t observations of n-dimensional centered feature vectors, $X_{i:}$, and aim to recover a d-dimensional re-representation, a $t \times d$ matrix Z, of the data ($d < n$). The difference is that we now consider to exploit an additional $t \times k$ indicator matrix, Y, with each row to indicate the class label for each observation $X_{i:}$; thus $\sum_{j=1}^{k} Y_{ij} = 1$. Since the label information Y is exploited in the latent low dimensional space discovery process, this is called supervised dimensionality reduction. For recovering Z, a key restriction that one would like to enforce is that the features used for coding, $Z_{:j}$, should be linearly independent; that is, one would like to enforce the constraint $Z^\top Z = I$, which ensures that the codes are expressed by orthogonal features in the low dimensional representation.

Given the above setup, in this paper, we are attempting to address the problem of supervised dimensionality reduction using a probabilistic latent variable model illustrated in Figure 1. Our

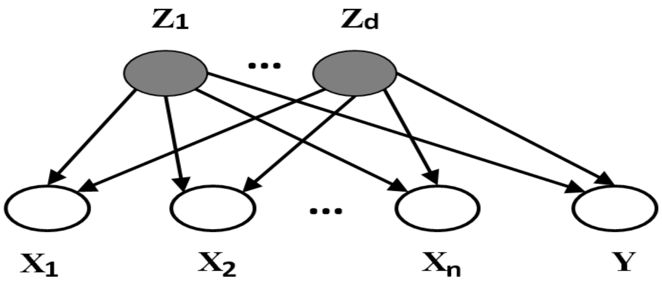

Figure 1: Illustration of the Latent Variable Model

intuition is that the important intrinsic structure (underlying feature representation) of the data should be able to accurately generate/predict the original data features and labels.

In this section, we formulate the low-dimensional principal component discovering problem as a conditional likelihood maximization problem based on exponential family model representations, which can be reformulated into a nondifferentiable convex optimization problem. We then exploit a sample-based approximation to unify exponential family models for different data types.

3.1 Convex Formulation of Supervised Exponential Family PCA

As with the generalized exponential family PCA [4], we attempt to find low-dimensional representation by maximizing the conditional likelihood of the observation matrix X and Y given the latent matrix Z, $\log P(X,Y|Z) = \log P(X|Z) + \log P(Y|Z)$. We assume $P(X|Z)$ is an exponential family distribution. Since the class variable y is discrete, thus $\log P(Y|Z)$ is simply the likelihood over a multi-class logistic regression problem, which can be formulated in a parametric form with parameters $\Omega, \mathbf{b}$

$$\log P(Y|Z,\Omega,\mathbf{b}) = \mathrm{tr}\left(Z\Omega Y^\top\right) + \mathbf{1}^\top Y\mathbf{b} - \sum_i A(Z_{i:},\Omega,\mathbf{b}) \tag{3}$$

where

$$A(Z_{i:},\Omega,\mathbf{b}) \quad = \quad \log\sum_{\ell=1}^{k}\exp\left(Z_{i:}\Omega\mathbf{1}_\ell + \mathbf{1}_\ell^\top\mathbf{b}\right) \tag{4}$$

is the log normalization function to ensure valid probability distribution; $\mathbf{b}$ is a bias vector; and $\mathbf{1}_\ell$ denotes a zero vector with a single 1 in the ℓth entry. Using the general exponential family representation in (1), and the multi-class logistic regression formulation in (3), a regularized version of the conditional likelihood maximization problem can be formulated as

$$\max_{Z:Z^\top Z=I}\max_{W,\Omega,\mathbf{b}} \log P(X|Z,W) - \frac{\beta}{2}\mathrm{tr}\left(WW^\top\right) + \log P(Y|Z,\Omega,\mathbf{b}) - \frac{\beta}{2}\left(\mathrm{tr}\left(\Omega\Omega^\top\right) + \mathbf{b}^\top\mathbf{b}\right)$$

$$
\begin{aligned}
= \max_{Z:Z^\top Z=I} \max_{W,\Omega,\mathbf{b}} \; & \mathrm{tr}\left(ZWX^\top\right) - \sum_i \left(A(Z_{i:}W) - \log P_0(X_{i:})\right) - \frac{\beta}{2}\mathrm{tr}\left(WW^\top\right) \\
& + \mathrm{tr}\left(Z\Omega Y^\top\right) + \mathbf{1}^\top Y\mathbf{b} - \sum_i A(Z_{i:},\Omega,\mathbf{b}) - \frac{\beta}{2}\left(\mathrm{tr}\left(\Omega\Omega^\top\right) + \mathbf{b}^\top\mathbf{b}\right)
\end{aligned} \tag{5}
$$

where W is a $d \times n$ parameter matrix for conditional model $P(X|Z)$; Ω is a $d \times k$ parameter matrix for conditional model $P(Y|Z)$ and $\mathbf{b}$ is a $k \times 1$ bias vector; $\mathbf{1}$ denotes the vector of all 1s; $A(Z_{i:}W)$ and $A(Z_{i:}, \Omega, \mathbf{b})$ are the log normalization functions defined in (2) and (4) respectively; β is a user provided parameter used to control the tradeoff between the likelhood terms and the the $L2$ regularization terms over the parameters $W, \Omega, \mathbf{b}$.

Theorem 1 *The optimization problem (5) is equivalent to*

$$
\begin{aligned}
\min_{U^x,U^y} \max_{M:I\succeq M\succeq 0,\; tr(M)=d} \quad & \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \frac{1}{2\beta} tr\left((X-U^x)(X-U^x)^\top M\right) \\
& + \sum_i A^*(U^y_{i:}) + \frac{1}{2\beta} tr\left((Y-U^y)(Y-U^y)^\top (M+E)\right)
\end{aligned} \tag{6}
$$

under the assumption that there are no ties between the dth and the $(d+1)$th largest eigenvalues of $(X-U^{x*})(X-U^{x*})^\top + (Y-U^{y*})(Y-U^{y*})^\top$, where U^{x*} and U^{y*} denote the optimal solution of (6); otherwise, (6) forms an upper bound for (5). Here E is a $t \times t$ matrix with all 1s; U^x is a $t \times n$ matrix; U^y is a $t \times k$ matrix; $A^*(U^x_{i:})$ and $A^*(U^y_{i:})$ are the Fenchel conjugates of $A(Z_{i:}W)$ and $A(Z_{i:}, \Omega, \mathbf{b})$ respectively. The primal solution $\{Z^*, W^*, \Omega^*, \mathbf{b}^*\}$ to (5) can be recovered by setting Z^* as the top d eigenvectors of the optimal M^* and then obtaining the model parameters $W^*, \Omega^*, \mathbf{b}^*$ by

$$
W^* = \frac{1}{\beta} Z^{*\top}(X - U^{x*}), \quad \Omega^* = \frac{1}{\beta} Z^{*\top}(Y - U^{y*}), \quad \mathbf{b}^* = \frac{1}{\beta}(Y - U^{y*})^\top \mathbf{1}
$$

Note that (6) is a min-max optimization problem. Moreover, for each fixed M, the outer minimization problem is obviously convex, since the Fenchel conjugates

$$
\begin{aligned}
A^*(U^x_{i:}) &= \max_W \; \mathrm{tr}\left(Z_{i:}WU^{x\top}_{i:}\right) - A(Z_{i:}W), \\
A^*(U^y_{i:}) &= \max_{\Omega,\mathbf{b}} \; \mathrm{tr}\left(\left(Z_{i:}\Omega + \mathbf{b}^\top\right) U^{y\top}_{i:}\right) - A(Z_{i:}, \Omega, \mathbf{b})
\end{aligned}
$$

are convex functions of U^x and U^y respectively [2]; that is, the objective function for the outer minimization is a pointwise supremum over an infinite set of convex functions. Thus the overall min-max optimization is convex [3], but apparently not necessarily differentiable. We will address the nondifferentiable training issue in Section 4.

The proof for Theorem 1 comprises of a series of reformulations based on standard results. We state these key reformulations using Lemmas. First, we derive the dual formulation of the inner maximization of (5).

Lemma 1 *The maximization problem*

$$\max_{W}\ tr\left(ZWX^\top\right) - \sum_i \left(A(Z_{i:}W) - \log P_0(X_{i:})\right) - \frac{\beta}{2} tr\left(WW^\top\right) \tag{7}$$

is equivalent to the following dual minimization

$$\min_{U^x}\ \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \frac{1}{2\beta} tr\left((X - U^x)(X - U^x)^\top ZZ^\top\right) \tag{8}$$

Proof: Note the log normalization function $A(Z_{i:}W)$ defined in (2) is convex in W and can be reexpressed as

$$A(Z_{i:}W) \quad = \quad \max_{U^x_{i:}} \mathrm{tr}\left(Z_{i:}WU^{x\top}_{i:}\right) - A^*(U^x_{i:})$$

where A^* is the Fenchel conjugate of A, a closed convex function [3, 20]. Thus, we can rewrite (7) as

$$\max_{W}\min_{U^x}\ \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \mathrm{tr}\left(ZW(X - U^x)^\top\right) - \frac{\beta}{2}\mathrm{tr}\left(WW^\top\right) \tag{9}$$

Let $F(W, U^x)$ denote the objective in (9). Crucially, one can verify that F satisfies the conditions of the strong min-max property [14, Theorem 37.3] and [2, Page 95], which allows the order of the minimization and maximization to be reversed. That is, (9) is equivalent to

$$\min_{U^x}\max_{W}\ \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \mathrm{tr}\left(ZW(X - U^x)^\top\right) - \frac{\beta}{2}\mathrm{tr}\left(WW^\top\right) \tag{10}$$

Now, since the objective function for the inner maximization on W is concave in W for fixed U^x, it can be solved by setting $d/dW = Z^\top(X - U) - \beta W = 0$, which implies $W = \frac{1}{\beta}Z^\top(X - U)$. Substituting this into (10) yields the result (8).

Lemma 2 *The maximization problem*

$$\max_{\Omega,\mathbf{b}}\ tr\left(Z\Omega Y^\top\right) + \mathbf{1}^\top Y\mathbf{b} - \sum_i A(Z_{i:}, \Omega, \mathbf{b}) - \frac{\beta}{2}\left(tr\left(\Omega\Omega^\top\right) + \mathbf{b}^\top\mathbf{b}\right) \tag{11}$$

is equivalent to the following dual minimization problem

$$\min_{U^y}\ \sum_i A^*(U^y_{i:}) + \frac{1}{2\beta} tr\left((Y - U^y)(Y - U^y)^\top(ZZ^\top + E)\right) \tag{12}$$

Proof: The proof is similar as above. The log normalization function $A(Z_{i:}, \Omega, \mathbf{b})$ can be reexpressed using its Fenchel conjugate dual; that is

$$A(Z_{i:}, \Omega, \mathbf{b}) \;=\; \max_{U^y_{i:}} \ \mathrm{tr}\left(\left(Z_{i:}\Omega + \mathbf{b}^\top\right) U^{y\top}_{i:}\right) - A^*(U^y_{i:})$$

Substituting this into (11) and then reversing the order of minimization and maximization according to the strong min-max property yields

$$\min_{U^y} \max_{\Omega, \mathbf{b}} \sum_i A^*(U^y_{i:}) + \mathrm{tr}\left(Z\Omega(Y - U^y)^\top\right) + \mathbf{1}^\top(Y - U^y)\mathbf{b} - \frac{\beta}{2}\left(\mathrm{tr}\left(\Omega\Omega^\top\right) + \mathbf{b}^\top\mathbf{b}\right) \tag{13}$$

The inner maximization function is apparently concave in Ω and $\mathbf{b}$, and thus can be solved by setting $d/d\Omega = Z^\top(Y - U^y) - \beta\Omega = 0$, $d/d\mathbf{b} = (Y - U^y)^\top\mathbf{1} - \beta\mathbf{b} = 0$, which yields $\Omega = \frac{1}{\beta}Z^\top(Y - U^y)$ and $\mathbf{b} = \frac{1}{\beta}(Y - U^y)^\top\mathbf{1}$ respectively. Substituting them into (13) yields the result (12).

Lemma 3 *The optimization problem (5) is equivalent to*

$$\max_{Z:Z^\top Z=I} \min_{U^x, U^y} \quad \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \frac{1}{2\beta} tr\left((X - U^x)(X - U^x)^\top ZZ^\top\right)$$
$$+ \sum_i A^*(U^y_{i:}) + \frac{1}{2\beta} tr\left((Y - U^y)(Y - U^y)^\top(ZZ^\top + E)\right) \tag{14}$$

Proof: The proof is simply a summarization of Lemma 1 and Lemma 2.

Next, we turn to the outer maximization over Z, which involves a non-convex constraint. Note that Z only appears in the form of inner product $ZZ^\top$ in (14). This allows us to rewrite the objective function in terms of a square matrix $M = ZZ^\top$. Although this is a relaxation of the original optimization problem in general, we can show later that the optimal solution can be preserved under an additional assumption.

Lemma 4 *The optimization (14) is upper bounded by*

$$\max_{M:I\succeq M\succeq 0,\ tr(M)=d} \min_{U^x, U^y} \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \frac{1}{2\beta} tr\left((X - U^x)(X - U^x)^\top M\right)$$
$$+ \sum_i A^*(U^y_{i:}) + \frac{1}{2\beta} tr\left((Y - U^y)(Y - U^y)^\top(M + E)\right) \tag{15}$$

Proof: The proof is based on the relationships hold between the following sets of constraints on M

$$\begin{aligned} & \{M : M = ZZ^\top \text{ for some } Z \text{ such that } Z^\top Z = I\} \\ = \; & \{M : I \succeq M \succeq 0, \mathrm{tr}(M) = d, M^2 = M\} \\ \subseteq \; & \{M : I \succeq M \succeq 0, \mathrm{tr}(M) = d\}. \end{aligned}$$

The first equality holds because both sets of constraints bound the eigenvalues of the matrices to be either 0 or 1, with exactly d of them being 1 [12]. Unfortunately, neither of the first two sets of constraints is convex on M due to that they contain quadratic equality constraints $M = ZZ^\top$ and $M^2 = M$ respectively. Nevertheless we can relax the second set of constraints merely by dropping the non-convex constraint $M^2 = M$, which then means the eigenvalues of M are only constrained to be between 0 and 1 with their sum totalling to d. The relaxed set of constraints is convex. Obviously since the problem is a maximization, an upper bound is obtained by relaxing the constraint.

Thus we have achieved a formulation where the problem is convex, albeit with a relaxation. That is, the outer maximization in (15) is concave in M, since a minimum of linear functions is concave and the constraints are convex [3]. Once again, one can further reverse the order of minimization and maximization based on strong min-max property of the objective function of (15).

Lemma 5 *The optimization problem (15) is equivalent to (6).*

Proof: The proof is immediate upon verifying the objective function of (15) satisfies the conditions of strong min-max property [14, Theorem 37.3].

So far we have actually shown (6) is an upper bound relaxation of (5). Below we will further establish this upper bound is as tight as being equivalent to the original problem (5) if the following condition is satisfied by the optimal solution (U^{x*}, U^{y*}, M^*) of (6)

$$\lambda_d(D) > \lambda_{d+1}(D), \text{ for } D = (X - U^{x*})(X - U^{x*})^\top + (Y - U^{y*})(Y - U^{y*})^\top \tag{16}$$

where $\lambda_d(D)$ denotes the dth largest eigenvalue of D.

Note that the inner maximization of (6) is in the form of a standard semidefinite program. We invoke a fundamental theorem about semidefinite programs of this form to achieve the following result.

Lemma 6 *The optimization problem (6) is equivalent to*

$$\begin{aligned} \min_{U^x, U^y} \max_{Z : Z^\top Z = I} \quad & \sum_i \left(A^*(U^x_{i:}) + \log P_0(X_{i:})\right) + \frac{1}{2\beta} tr\left((X - U^x)(X - U^x)^\top ZZ^\top\right) \\ & + \sum_i A^*(U^y_{i:}) + \frac{1}{2\beta} tr\left((Y - U^y)(Y - U^y)^\top (ZZ^\top + E)\right) \end{aligned} \tag{17}$$

Moreover, under the assumption that the optimal solution (U^{x*}, U^{y*}, M^*) of (6) satisfies the condition (16), we have $M^* = Z^* Z^{*\top}$, where Z^* is the optimal solution for (17), which implies $rank(M^*) = d$.

Proof: The proof is based on a standard result from [11], which shows that the semidefinite program

$$\max_{M: I \succeq M \succeq 0, \mathrm{tr}(M)=d} \mathrm{tr}(MA)$$

is equivalent to solving

$$\max_{Z: Z^\top Z = I} \mathrm{tr}(Z^\top A Z)$$

and when there are no ties between the dth and $(d+1)$th largest eigenvalues of A, that is $\lambda_d(A) > \lambda_{d+1}(A)$, the primal and dual solutions are connected by $M^* = Z^* Z^{*\top}, Z^* = Q^{(d)}_{max}(A)$, where $Q^d_{max}(A)$ denotes the matrix formed by the top d eigenvectors of A.

Lemma 7 *The optimization problem (6) is equivalent to (14) when its solution satisfies the condition (16).*

Proof: Let (U^{x*}, U^{y*}, M^*) be an optimal solution to (6). When the optimization problem (6) satisfies the condition (16), we have $M^* = Z^* Z^{*\top}$ according to Lemma 6, where Z^* belongs to the optimal solution of (17); or equivalently $rank(M^*) = d$. Note that (6) is an equivalent optimization problem to (15) by Lemma 5. Hence we have that (U^{x*}, U^{y*}, M^*) is an optimal solution of (15). Recall that (15) was an upper bound on (14) only insofar as the constraint $M^2 = M$ was dropped. However, the solution M^* to (15) automatically satisfies $M^{*2} = M^*$. Hence it also is a solution of (14).

Finally, Theorem 1 can be proved by summarizing all the lemmas established above.

3.2 Sample-based Approximation

In the previous section, we have formulated our supervised exponential family PCA as a convex optimization problem (6). However, before attempting to devise a training algorithm to solve it, we have to provide some concrete forms for the Fenchel conjugate functions $A^*(U^x_{i:})$ and $A^*(U^y_{i:})$. For different exponential family models, the Fenchel conjugate functions A^* are different; see [20, Table 2]. For example, since the y variable in our model is a discrete class variable, it takes a multinomial distribution. Thus the Fenchel conjugate function $A^*(U^y_{i:})$ is given by

$$A^*(U^y_{i:}) \; = \; A^*(\Theta^y_{i:}) \; = \; \mathrm{tr}\left(\Theta^y_{i:} \log \Theta^{y\top}_{i:}\right), \text{ where } \Theta^y \geq 0, \; \Theta^y \mathbf{1} = \mathbf{1} \qquad (18)$$

The specific exponential family model is determined by the data type and distribution. PCA and SPPCA use Gaussian models, thus their performances might be degraded when the data distribution is non-Gaussian. However, it is tedious and sometimes hard to choose the most appropriate exponential family model to use for each specific application problem. Moreover, the log normalization function A and its Fenchel conjugate A^* might not be easily computable. For these reasons, we propose to use a sample-based approximation to the integral (2) and achieve an empirical approximation to the true underlying exponential family model as follows. If one replaces the integral definition (2) with an empirical definition, $A(Z_{i:}W) = \log \sum_j \exp\left(Z_{i:}WX_{j:}^\top\right)/t$, then the conjugate function can be given by

$$A^*(U^x_{i:}) \;=\; A^*(\Theta^x_{i:}) \;=\; \operatorname{tr}\left(\Theta^x_{i:} \log \Theta^{x\top}_{i:}\right) - \log(1/t), \;\text{ where }\; \Theta^x \geq 0,\; \Theta^x\mathbf{1} = \mathbf{1} \tag{19}$$

With this sample-based approximation, problem (6) can be expressed as

$$\begin{aligned}\min_{\Theta^x,\Theta^y}\; \max_{M: I\succeq M\succeq 0,\, \operatorname{tr}(M)=d} \quad & \operatorname{tr}\left(\Theta^x \log \Theta^x\right) + \frac{1}{2\beta}\operatorname{tr}\left((I-\Theta^x)K(I-\Theta^x)^\top M\right) && (20)\\ & + \operatorname{tr}\left(\Theta^y \log \Theta^y\right) + \frac{1}{2\beta}\operatorname{tr}\left((Y-\Theta^y)(Y-\Theta^y)^\top (M+E)\right) \\ \text{subject to} \quad & \Theta^x \geq 0,\; \Theta^x\mathbf{1}=\mathbf{1};\; \Theta^y \geq 0,\; \Theta^y\mathbf{1}=\mathbf{1} && (21)\end{aligned}$$

One benefit of working with this sample-based approximation is that it is automatically kernelized, $K = XX^\top$, to enable non-linearity to be conveniently introduced.

4 Efficient Global Optimization

The optimization (20) we derived in the previous section is a convex-concave min-max optimization problem. The inner maximization of (20) is a well known problem with a closed-form solution [11]. Let $D = (I-\Theta^x)K(I-\Theta^x)^\top + (Y-\Theta^y)(Y-\Theta^y)^\top$. If $\lambda_d(D) > \lambda_{d+1}(D)$, the inner maximization of (20) has a closed-form solution

$$M^* = Z^* Z^{*\top} \quad \text{and} \quad Z^* = Q^d_{max}(D).$$

If $\lambda_1(D) \geq \cdots \geq \lambda_r(D) > \lambda_{r+1}(D) = \cdots = \lambda_{r+m}(D) > \lambda_{r+m+1}(D) \geq \cdots \geq \lambda_t(D)$ for $r+1 \leq d < r+m \leq t$, the inner maximization has closed-form solutions

$$\{M^* = Z_1 Z_1^\top + Z_2 R Z_2^\top\}$$

for $Z_1 = Q^r_{max}(D)$, $Z_2 = Q^{r+1:r+m}_{max}(D)$, which denote the matrices formed by the top r eigenvectors of D and the top $(r+1)$th to $(r+m)$th eigenvectors of D respectively; and R is a $m \times m$ symmetric matrix such that $I \succeq R \succeq 0, \operatorname{tr}(R) = d - r$.

However, the overall outer minimization problem is nondifferentiable with respect to Θ^x and Θ^y. Thus the standard first-order or second-order optimization techniques that rely on the standard gradients can not be applied here. In this section, we deploy a bundle method to solve this nondifferentiable min-max optimization.

4.1 Bundle Method for Min-Max Optimization

The bundle method is an efficient subgradient method for nondifferentiable convex optimization; it relies on the computation of subgradient terms of the objective function. A vector $\mathbf{g}$ is a **subgradient** of function f at point $\mathbf{x}$, if $f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y}-\mathbf{x}), \forall \mathbf{y}$. To adapt standard bundle methods to our specific min-max problem, we need to first address the critical issue of subgradient computation.

Proposition 1 *Consider a joint function $h(\mathbf{x},\mathbf{y})$ defined over $\mathbf{x} \in \mathcal{X}$ and $\mathbf{y} \in \mathcal{Y}$, satisfying: (1) $h(\cdot,\mathbf{y})$ is closed and convex for all $\mathbf{y} \in \mathcal{Y}$; (2) $h(\mathbf{x},\cdot)$ is closed and concave for all $\mathbf{x} \in \mathcal{X}$, and $\sup_{\mathbf{y}\in\mathcal{Y}} h(\mathbf{x},\mathbf{y}) < \infty$ for all $\mathbf{x}$. Let $f(\mathbf{x}) = \max_{\mathbf{y}} h(\mathbf{x},\mathbf{y})$, and $q(\mathbf{x}_0) = \arg\max_{\mathbf{y}} h(\mathbf{x}_0,\mathbf{y})$. Assume that $\mathbf{g}$ is a gradient of $h(\cdot,q(\mathbf{x}_0))$ at $\mathbf{x}=\mathbf{x}_0$, then $\mathbf{g}$ is a subgradient of $f(\mathbf{x})$ at $\mathbf{x}=\mathbf{x}_0$.*

Proof:

$$\begin{aligned} f(\mathbf{x}) &= \max_{\mathbf{y}} h(\mathbf{x},\mathbf{y}) \ \geq \ h(\mathbf{x},q(\mathbf{x}_0)) \\ &\geq h(\mathbf{x}_0,q(\mathbf{x}_0)) + \mathbf{g}^\top(\mathbf{x}-\mathbf{x}_0) \quad \text{(since } h(\cdot,\mathbf{y}) \text{ is convex for all } \mathbf{y}\in\mathcal{Y}) \\ &= f(\mathbf{x}_0) + \mathbf{g}^\top(\mathbf{x}-\mathbf{x}_0) \quad \text{(by the definitions of } f(\mathbf{x}) \text{ and } q(\mathbf{x}_0)) \end{aligned}$$

Thus $\mathbf{g}$ is a subgradient of $f(\mathbf{x})$ at $\mathbf{x}=\mathbf{x}_0$ according to the definition of subgradient.

According to Proposition 1, the subgradients of our outer minimization objective function f in (20) over Θ^x and Θ^y can be given by

$$\partial_{\Theta^x} f \ni (\log\Theta^x + 1 - \frac{1}{\beta}M^*(I-\Theta^x)K), \ \ \partial_{\Theta^y} f \ni (\log\Theta^y + 1 - \frac{1}{\beta}M^*(Y-\Theta^y)) \qquad (22)$$

where M^* is the optimal inner maximization solution at the current point $[\Theta^x,\Theta^y]$.

Algorithm 1 illustrates the bundle method we developed to solve the infinite min-max optimization (20), where the linear constraints (21) over Θ^x and Θ^y can be conveniently incorporated into the quadratic bound optimization. One important issue in this algorithm is how to manage the size of the linear lower bound constraints formed from the active set $\mathbf{B}$ (defined in Algorithm 1), as it incrementally increases with new points being explored. To solve this problem, we noticed the Lagrangian dual parameters $\boldsymbol{\alpha}$ for the lower bound constraints obtained by the quadratic optimization in step 1 is a sparse vector, indicating that many lower bound constraints can be turned off. Moreover, any constraint that is turned off will mostly stay off in the later steps. Therefore, for the bundle method we developed, whenever the size of $\mathbf{B}$ is larger than a given constant b, we will keep the active points of $\mathbf{B}$ that correspond to the first b largest $\boldsymbol{\alpha}$ values, and drop the remaining ones.

Algorithm 1 Bundle Method for Min-Max Optimization in (20)

```
Input: $\bar{\delta} > 0, m \in (0,1)$, $b \in I\!N$, $\mu \in I\!R$
Initial: Find an initial point $\theta^*$ satisfying the linear constraints in (21); compute $f(\theta^*)$.
        Let $\ell = 1, \theta^\ell = \theta^*$, compute $\mathbf{g}^\ell \in \partial_{\theta^\ell} f$ by (22); $e^\ell = f(\theta^*) - f(\theta^\ell) - \mathbf{g}^{\ell\top}(\theta^* - \theta^\ell)$.
        Let $\mathbf{B} = \{(e^\ell, \mathbf{g}^\ell)\}$, $\hat{\varepsilon} = \mathrm{Inf}, \hat{\mathbf{g}} = \mathbf{0}$; $\ell = \ell + 1$.
repeat
  1. Solve quadratic minimization for solution $\hat{\theta}$, and Lagrangian dual parameters $\boldsymbol{\alpha}$ w.r.t. the lower bound linear constraints in $\mathbf{B}$ [1]:
        $\hat{\theta} = \arg\min_{\theta} \psi_\ell(\theta) + \frac{\mu}{2}\|\theta - \theta^*\|^2$, subject to the linear constraints in (21)
        where $\psi_\ell(\theta) = f(\theta^*) + \max\{\, -\hat{\varepsilon} + \hat{\mathbf{g}}^\top(\theta - \theta^*),\ \max_{(e^i, \mathbf{g}^i) \in \mathbf{B}} \{-e^i + \hat{\mathbf{g}}^{i\top}(\theta - \theta^*)\}\}$
  2. Define $\delta_\ell = f(\theta^*) - [\psi_\ell(\hat{\theta}) + \frac{\mu}{2}\|\hat{\theta} - \theta^*\|^2 \geq 0$. If $\delta_\ell < \hat{\delta}$, return.
  3. Conduct line search to minimize $f(\theta^\ell)$ with $\theta^\ell = \gamma\theta^* + (1-\gamma)\hat{\theta}$, for $0 < \gamma < 1$.
  4. Compute $\mathbf{g}^\ell \in \partial_{\theta^\ell} f$ by (22); $e^\ell = f(\theta^*) - f(\theta^\ell) - \mathbf{g}^{\ell\top}(\theta^* - \theta^\ell)$; update $\mathbf{B} = \mathbf{B} \cup \{(e^\ell, \mathbf{g}^\ell)\}$.
  5. If $f(\theta^*) - f(\theta^\ell) \geq m\delta_\ell$, then take a serious step:
        (1) update: $e^i = e^i + f(\theta^\ell) - f(\theta^*) + \mathbf{g}^{i\top}(\theta^* - \theta^\ell)$;
        (2) update the aggregation: $\hat{\mathbf{g}} = \sum_i \alpha_i \mathbf{g}^i$, $\hat{\varepsilon} = \sum_i \alpha_i e^i$;
        (3) update the stored solution: $\theta^* = \theta^\ell, f(\theta^*) = f(\theta^\ell)$.
  6. If $|\mathbf{B}| > b$, reduce $\mathbf{B}$ set according to $\boldsymbol{\alpha}$.
  7. $\ell = \ell + 1$.
until maximum iteration number is reached
```

4.2 Coordinate Descent Procedure

An important factor affecting the running efficiency is the size of the problem. The convex optimization (20) works in the dual parameter space, where the size of the parameters $\Theta = \{\Theta^x, \Theta^y\}$, $t \times (t+k)$, depends only on the number of training samples, t, not on the feature size, n. For high dimensional small data sets ($n \gg t$), our dual optimization is certainly a good option. However, with the increase of t, our problem size will increase in an order of $O(t^2)$. It might soon become too large to handle for the quadratic optimization step of the bundle method.

On the other hand, the optimization problem (20) possesses a nice semi-decomposable structure: one equality constraint in (21) involves only one row of the Θ; that is, the Θ can be separated into rows without affecting the equality constraints. Based on this observation, we develop a coordinate descent procedure to obtain scalability of the bundle method over large data sets. Specifically, we put an outer loop above the bundle method. Within each of this outer loop iteration, we randomly separate the Θ parameters into m groups, with each group containing a subset rows of Θ; and we then use bundle method to sequentially optimize each subproblem defined on one group of Θ parameters while keeping the remaining rows of Θ fixed. Although coordinate descent with a nondifferentiable convex objective is not guaranteed to converge to a minimum in general [19], we have found that this procedure performs quite well in practice, as shown in the experimental results.

5 Projection for Testing Data

One important issue for supervised dimensionality reduction is to map new testing data into the dimensionality-reduced principal dimensions. We deploy a simple procedure for this purpose. After training, we obtain a low-dimensional representation Z for X, where Z can be viewed as a linear projection of X in some transformed space $\psi(X)$ through a parameter matrix U; such that $Z = \psi(X)U = \psi(X)\psi(X)^\top K^+\psi(X)U$, where K^+ denotes the pseudo inverse of $K = \psi(X)\psi(X)^\top$. Then a new testing sample $\mathbf{x}^*$ can be projected by

$$\mathbf{z}^* = \psi(\mathbf{x}^*)\psi(X)^\top K^+\psi(X)U = k(\mathbf{x}^*, X)K^+Z \tag{23}$$

6 Experimental Results

In order to evaluate the performance of the proposed supervised exponential family PCA (SEPCA) approach, we conducted experiments over both synthetic and real data, and compared to supervised dimensionality reduction with generalized linear models (SDR_GLM), supervised probabilistic PCA (SPPCA), linear discriminant analysis (LDA), and colored maximum variance unfolding (MVU). The projection procedure (23) is used for colored MVU as well. In all the experiments, we used $\mu = 1$ for Algorithm 1, and used $\alpha = 0.0001$ for SDR_GLM as suggested in [13].

6.1 Experiments on Synthetic Data

Two synthetic experiments were conducted to compare the five approaches under controlled conditions. The first synthetic data set is formed by first generating four Gaussian clusters in a two-dimensional space, with each corresponding to one class, and then adding the third dimension to each point by uniformly sampling from a fixed interval. This experiment attempts to compare the performance of the five approaches in the situation where the data distribution does not satisfy the Gaussian assumption. The left figure in Figure 2 shows the original 3-dimensional test data. Figure 3 shows the projection results for each approach in a two dimensional space for 120 testing points after being trained on a set with 80 points. In this case, SEPCA and LDA outperform all the other three approaches.

The second synthetic experiment is designed to test the capability of performing nonlinear dimensionality reduction. The synthetic data is formed by first generating two circles in a two dimensional space (one circle is located inside the other one), with each circle corresponding to one class, and then the third dimension sampled uniformly from a fixed interval. As SDR_GLM does not provide a nonlinear form, we conducted the experiment with only the remaining four

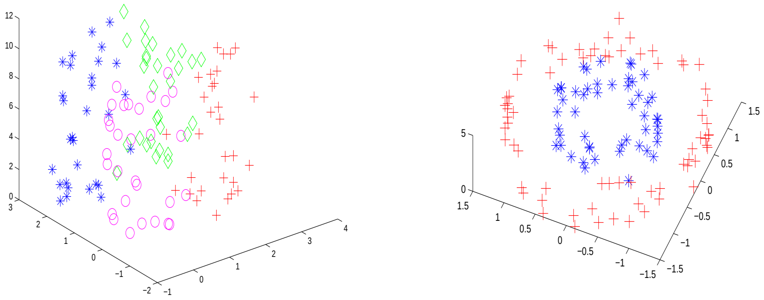

Figure 2: The original 3-dimensional test data: the left figure is for synthetic experiment 1 and and the right figure is for synthetic experiment 2. Each color indicates one class.

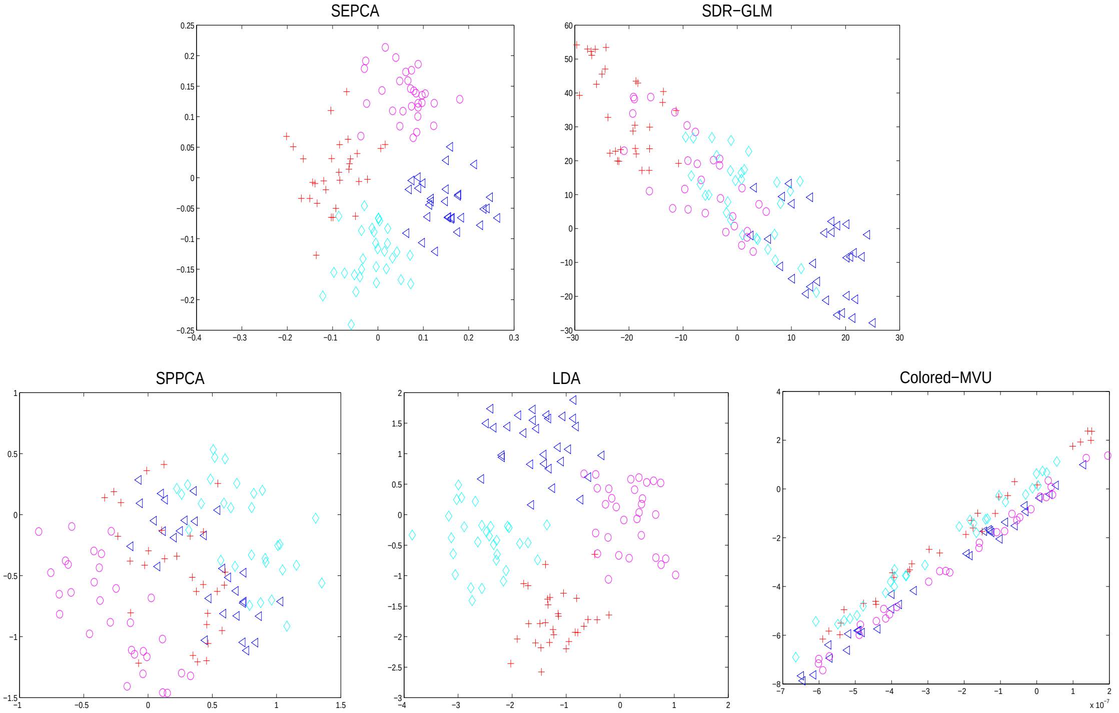

Figure 3: Projection results on test data for synthetic experiment 1. Each color indicates one class.

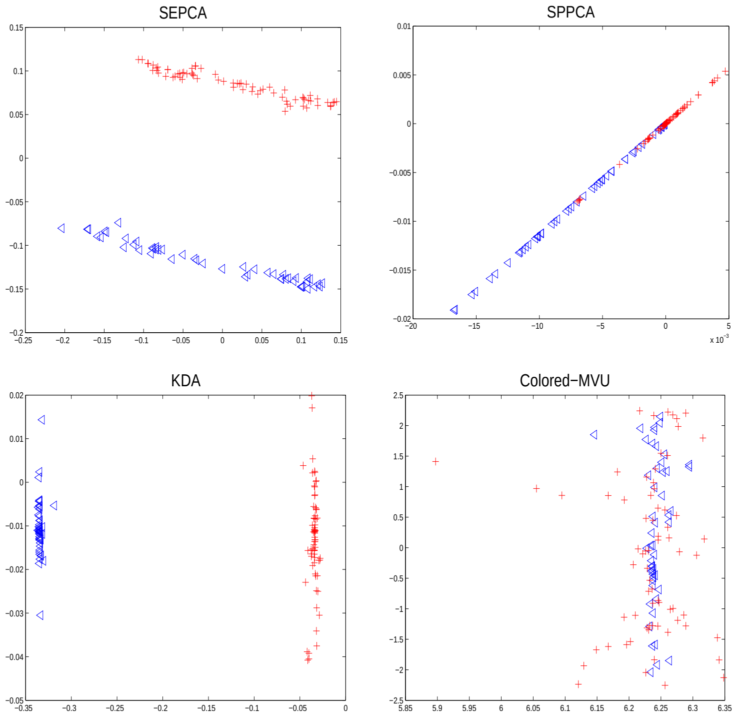

Figure 4: Projection results on test data for synthetic experiment 2. Each color indicates one class.

Table 1: Data set information and test accuracy results (%)

						SDR_			colored
Dataset	#Data	#Dim	#Class	FULL	SEPCA	GLM	SPPCA	LDA	MVU
Yale	165	4096	15	65.3	64.4	58.8	51.6	31.0	21.1
YaleB	2414	1024	38	47.0	20.5	19.0	9.8	6.2	2.8
11 Tumor	174	12533	11	77.6	88.9	63.5	63.0	23.7	40.2
Usps3456	120	256	4	82.1	79.7	77.9	78.5	74.3	75.8
Newsgroup	19928	25284	20	32.1	16.9	–	6.9	10.0	10.4

approaches. For LDA, we used its kernel variant, KDA. A Gaussian kernel with $\sigma = 1$ was used for SEPCA, SPPCA and KDA. The right figure in Figure 2 shows the original 3-dimensional test data. Figure 4 shows the projection results for each approach in a two dimensional space for 120 testing points after being trained on a set with 95 points. Again, SEPCA and KDA achieve good class separations and outperform the other two approaches.

6.2 Experiments on Real Data

To better characterize the performance of dimensionality reduction in a supervised manner, we conducted some experiments on a few high dimensional multi-class real world data sets. The left side of Table 1 provides the information about these data sets.

Our experiments were conducted in the following way. We randomly selected 3∼5 examples from each class to form the training set and used the remaining examples as the test set. For each approach, we first learned the dimensionality reduction model on the training set. Moreover, we also trained a logistic regression classifier using the projected training set in the reduced low dimensional space. (Note, for SEPCA, a classifier was trained simultaneously during the process of dimensionality reduction optimization.) Then the test data were projected into the low dimensional space according to each dimensionality reduction model. Finally, the projected test set for each approach were classified using each corresponding logistic regression classifier. The right side of Table 1 shows the classification accuracies on the test set for each approach. To better understand the quality of the classification using projected data, we also included the standard classification results, indicated as ’FULL’, using the original high dimensional data. (Note, we are not able to obtain any result for SDR_GLM on the newsgroup data as it is inefficient for very high dimensional data.) The results reported here are averages over 20 repeated runs, and the projection dimension $d = 10$. Still the proposed SEPCA presents the best performance among the compared approaches. But different from the synthetic experiments, LDA does not work well on these real data sets.

We have also conducted experiments to compare the SEPCA with the other few approaches

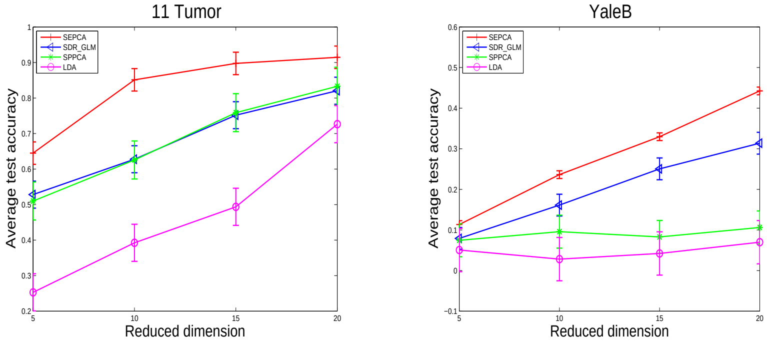

Figure 5: Results on the Tumor and YaleB data sets with a set of different reduced dimensions $d = 5, 10, 15, 20$.

using a set of different reduced (projection) dimensions: 5, 10, 15 and 20. Figure 5 shows the results on the Tumor data set and YaleB data set. With the increasing of the projection dimension, each approach obtains a better accuracy. But once again, the SEPCA shows a clear advantage over the other approaches.

The results on both synthetic and real data show that SEPCA outperforms the other four approaches. This might be attributed to its adaptive exponential family model approximation and its global optimization, while SDR_GLM and SPPCA apparently suffer from local optima.

7 Conclusions

In this paper, we propose a supervised exponential family PCA (SEPCA) approach, which can be solved efficiently to find global solutions. Moreover, SEPCA overcomes the limitation of the Gaussian assumption of PCA and SPPCA by using a data adaptive approximation for exponential family models. A simple, straightforward projection method for new testing data has also been constructed. Empirical study suggests that this SEPCA outperforms other supervised dimensionality reduction approaches, such as SDR_GLM, SPPCA, LDA and colored MVU.

References

[1] A. Belloni. Introduction to bundle methods. Technical report, MIT, 2005.

[2] J. Borwein and A. Lewis. *Convex Analysis and Nonlinear Optimization*. Springer, 2000.

[3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge U. Press, 2004.

[4] M. Collins, S. Dasgupta, and R. Schapire. A generalization of principal component analysis to the exponential family. In *Advances in Neural Information Processing Systems*, 2001.

[5] R. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[6] Y. Guo and D. Schuurmans. Efficient global optimization for exponential family PCA and low-rank matrix factorization. In *Allerton Conf. on Comm., Control, and Comput.*, 2008.

[7] I. Jolliffe. *Principal Component Analysis.* Springer Verlag, 2002.

[8] N. Lawrence. Probabilistic non-linear principle component analysis with gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1816, 2005.

[9] P. McCullagh and J. Nelder. *Generalized Linear Models.* Monographs on Statistics and Applied Probability. Chapman and Hall, 1983.

[10] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. Muller. Fisher discriminant analysis with kernels. In *IEEE Neural Networks for Signal Processing Workshop*, 1999.

[11] M. Overton and R. Womersley. Optimality conditions and duality theory for minimizing sums of the largest eigenvalues of symmetric matrices. *Math. Prog.*, 62:321–357, 1993.

[12] J. Peng and Y. Wei. Approximating k-means-type clustering via semidefinite programming. *SIAM Journal on Optimization*, 18(1):186–205, 2007.

[13] I. Rish, G. Grabarnilk, G. Cecchi, F. Pereira, and G. Gordon. Closed-form supervised dimensionality reduction with generalized linear models. In *Proceedings of International Conference on Machine Learning*, 2008.

[14] R. Rockafellar. *Convex Analysis.* Princeton Univ. Press, 1970.

[15] Sajama and A. Orlitsky. Semi-parametric exponential family PCA. In *Advances in Neural Information Processing Systems*, 2004.

[16] Sajama and A. Orlitsky. Supervised dimensionality reduction using mixture models. In *Proceedings of the International Conference on Machine Learning)*, 2005.

[17] L. Song, A. Smola, K. Borgwardt, and A. Gretton. Colored maximum variance unfolding. In *Advances in Neural Information Processing Systems*, 2007.

[18] M. Tipping and C. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, B*, 6(3):611–622, 1999.

[19] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory and Applications*, 109:457–494, 2001.

[20] M. Wainwright and M. Jordan. Graphical models, exponential families, and variational inference. Technical Report TR-649, UC Berkeley, Dept. Statistics, 2003.

[21] S. Yu, K. Yu, V. Tresp, H. Kriegel, and M. Wu. Supervised probabilistic principal component analysis. In *Proc. of 12th ACM SIGKDD International Conf. on KDD*, 2006.