Structured Latent Factor Analysis

Yunlong He
Georgia Institute of Technology
heyunlong@gatech.edu

Koray Kavukcuoglu
NEC Labs America
koray@nec-labs.com

Yanjun Qi
NEC Labs America
yanjun@nec-labs.com

Haesun Park
Georgia Institute of Technology
hpark@cc.gatech.edu

1 Introduction

Latent factor models (LFMs) are a set of unsupervised methods that model observed high-dimensional data examples by linear combination of latent factors. To enable efficient processing of large data collections, LFMs aim to find concise descriptions of the members of a data collection while preserving the essential statistical information which is useful for basic tasks such as classification, indexing or summarization. Due to its simple form and computation convenience, latent factor models have been very popular in modeling and analyzing massive data sets such as text documents and images [Hastie et al., 2001].

In this paper, our goal is to learn interpretable lower dimensional latent representations from a set of data samples and simultaneously model the relationship between latent factors. This is largely motivated by the massive-scale data corpora available online, and the urgent demands for understanding the hidden structure inside these high-dimensional data sets. It is very helpful to not only find the common hidden factors but also explore the structural relationship between these latent groups. For example, a piece of news text about ”budget spending” is much more likely to be about ”war”, compared to the ”entertainment” topics. These types of “positive correlated” or “negative correlated” relationships between latent topics will help us explore and visualize a large collection of documents much more deeply and in a more structured way.

In this paper, a method named “structured latent factor analysis” is proposed to simultaneously learn the latent factors and their pairwise relationships from data. Derived from probabilistic modeling of data, SLFA can be seen as a generalized matrix factorization task using a special regularization term. By modeling the distributions of sample embedding vectors via a Sparse Gaussian Graphical model, we discover the pairwise relationships between latent factors through SGGM's precision matrix. On multiple synthetic and real-world data sets, SLFA demonstrates its superiority over both classic and state-of-the-art methods.

2 Preliminaries

Latent Factor Models Latent factor models[1] study a random vector $\mathbf{x} \in \mathbb{R}^M$ by assuming that it is generated by a linear combination of a set of basis vectors, i.e.,

$$\mathbf{x} = \mathbf{B}\mathbf{s} + \epsilon = \mathbf{B}_1 s_1 + \cdots + \mathbf{B}_K s_K + \epsilon \tag{1}$$

[1] Throughout this paper, we abuse the term 'latent factors' to indicate the basis vectors $\mathbf{B}_1, \ldots, \mathbf{B}_K$, which is different from the convention that $s_1, \ldots, s_K$ are called factors and $\mathbf{B}_1, \ldots, \mathbf{B}_K$ are called the loading vectors.

where $\mathbf{B} = [\mathbf{B}_1, \dots, \mathbf{B}_K]$ stores the set of fixed but unknown basis and ϵ describes noise.

Given a set (with size N) of observations $\mathbf{x} \in \mathbb{R}^M$ from $\mathbf{X} = [\mathbf{X}_1, \dots, \mathbf{X}_N] \in \mathbb{R}^{M \times N}$, LFM could be generally formulated as a matrix factorization problem that minimizes the reconstruction error over the whole dataset with respect to $\mathbf{B} \in \mathbb{R}^{M \times K}$ and $\mathbf{S} \in \mathbb{R}^{K \times N}$:

$$\min \frac{1}{N} \|\mathbf{X} - \mathbf{B}\mathbf{S}\|_F^2 \tag{2}$$

where $\| \cdot \|_F$ is the matrix Frobenius norm and certain constraints or penalties normally apply on $\mathbf{B}$ and $\mathbf{S}$. Once the $\mathbf{B}$ and $\mathbf{S}$ are obtained, we can analyze the dataset by checking the meaning of the latent factors $\mathbf{B}_1, \dots, \mathbf{B}_K$ and use the K dimensional representation $\mathbf{S} = [\mathbf{S}_1, \dots, \mathbf{S}_N]$ of the original data $\mathbf{X}$ for further tasks such as classification.

If the sample mean of $\mathbf{X}$ is zero, principal component analysis (PCA) [Hastie et al., 2001] gives the optimal solution for problem (2), though PCA is usually formulated as eigenvalue problem for the sample covariance matrix $\hat{\mathbf{\Sigma}} = \frac{1}{N}\mathbf{X}^T\mathbf{X}$. One can also directly apply singular value decomposition (SVD) on the data matrix without the centering step. In information retrieval (IR) community, the method of computing the SVD of the document-term matrix is called Latent Semantic Analysis (LSA) [Deerwester et al., 1990] and produces a set of orthogonal topics. Other important variants include non-negative matrix factorization (NMF) [Lee and Seung, 1999], which imposes non-negativity upon $\mathbf{B}$ and $\mathbf{S}$; sparse coding or dictionary learning [Olshausen et al., 1996], which imposes sparsity on $\mathbf{S}$; and sparse LSA [Chen et al., 2011], which enforces sparsity on $\mathbf{B}$. Despite the popularity of these previous methods, none of them considers the relationship between the latent factors. In this sense, they are incapable of recovering the deeper structure of the dataset.

Sparse Gaussian Graphical Model A Gaussian graphical model [Jordan, 1998] characterizes the patterns of association among multiple variables that are jointly Gaussian. Zeros in the inverse covariance matrix (i.e. so-called precision matrix $\Phi = \Sigma^{-1}$) correspond to conditional independence properties among the variables. When given a sample covariance matrix $\hat{\mathbf{\Sigma}}$, a sparse Gaussian graphical model [Yuan and Lin, 2007] estimates $\mathbf{\Sigma}$ or $\mathbf{\Phi}$ by solving a MLE problem with an ℓ_1-norm penalty encouraging sparsity of precision matrix or conditional independence among variables:

$$\min_{\mathbf{\Phi} \succ 0} (-\log\det\mathbf{\Phi} + < \hat{\mathbf{\Sigma}}, \mathbf{\Phi} > + \rho\|\mathbf{\Phi}\|_1) \tag{3}$$

for some $\rho > 0$, where $\|\mathbf{\Phi}\|_1 = \sum_i \sum_j |\Phi_{i,j}|$.

When applying sparse Gaussian graphical model to very high dimensional data such as text, it is normally difficult to analyze the result, simply due to the huge size of the resulting graph. This obstacle could be partially tackled for cases where original variables act in the pattern of groups. In Section 3, we point out that our proposed SLFA could be treated as a generalization of sparse Gaussian graphical model which finds the associations between latent groups and therefore provides a smaller graph between factors which is much easier to analyze.

3 Structured Latent Factor Analysis

3.1 Formulation of SLFA

Assume that data sample $\mathbf{x}$ is drawn from the normal distributions, i.e.,

$$p(\mathbf{x}|\eta) = (2\pi)^{-M/2} exp(-\frac{1}{2\sigma^2}\|\mathbf{x} - \eta\|^2). \tag{4}$$

Let the natural parameter η represented by a linear combination of basis vectors $\eta = \mathbf{B}\mathbf{s}$ where $\mathbf{B}$ is the basis matrix. To model the relationship between latent factors, we impose a Gaussian prior distribution on the coefficient vector: $\mathbf{s} \sim N(\mathbf{0}, \mathbf{\Phi}^{-1})$, and use a sparsity-inducing prior for the precision matrix $exp(-\frac{1}{2}\rho_1\|\Phi\|_1)$ in order to encourage a parsimonious and less over-fitting model. Moreover, the sparse structure of $\mathbf{\Phi}$ will ease the analysis of the relational structure between the latent factors.

Given a set of observations of $\mathbf{x}$ which builds the data matrix $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_N]$, the posteriori is proportional to the product of likelihood function and prior distributions:

$$p(\mathbf{B}, \mathbf{S}, \mathbf{\Phi}|\mathbf{X}) \propto \prod_i \{exp(-\frac{1}{2\sigma^2}\|\mathbf{X}_i - \mathbf{B}\mathbf{S}_i\|^2) \times \frac{1}{\det(\mathbf{\Phi})^{\frac{1}{2}}} exp(-\frac{1}{2}\mathbf{S}_i^T\mathbf{\Phi}\mathbf{S}_i) \times exp(-\frac{1}{2}\rho\|\mathbf{\Phi}\|_1)\},$$

where $\rho = \frac{1}{N}\rho_1$ is a hyper-parameter. The Maximum a Posteriori (MAP) estimates of the basis matrix $\mathbf{B}$, the coefficient matrix $\mathbf{S}$ and the precision matrix $\mathbf{\Phi}$ are therefore the solution of the following optimization problem:

$$\begin{aligned} &\underset{\mathbf{B},\mathbf{S},\mathbf{\Phi}}{\text{minimize}} \ \frac{1}{N}\|\mathbf{X} - \mathbf{B}\mathbf{S}\|_F^2 + \sigma^2(\frac{1}{N}\text{tr}(\mathbf{S}^T\mathbf{\Phi}\mathbf{S}) - \log\det(\mathbf{\Phi}) + \rho\|\mathbf{\Phi}\|_1) \\ &\text{subject to } \mathbf{B} \geq \mathbf{0}, \|\mathbf{B}_k\| \leq 1, k = 1, \ldots, K, \\ &\qquad \mathbf{\Phi} \succcurlyeq 0. \end{aligned} \tag{5}$$

where additional constrains $\mathbf{B} \geq \mathbf{0}$ and $\|\mathbf{B}_k\| \leq 1$ are introduced for the identifiability of the model, and the constraint $\mathbf{\Phi} \succcurlyeq 0$ ensures the precision matrix $\mathbf{\Phi}$ is nonnegative-definite.

The first part of the objective function is a matrix factorization problem minimizing the reconstruction error. The second part is the sparse Gaussian Graphical Model problem. Therefore, the first part emphasizes reconstructing the data by the latent factors, while the second part emphasizes our prior assumption that the coefficient vector $\mathbf{s}$ follows a Gaussian distribution with a sparse precision matrix. The two parts of the objective function will compete to explain the observed data.

If the parameter $\sigma^2 = 0$ (i.e., non-informative prior), then problem (5) is a semi-nonnegative matrix factorization problem with nonnegativity constraint on $\mathbf{B}$. If $\sigma^2 > 0$ and $\mathbf{\Phi}$ is fixed, then problem (5) with respect to $\mathbf{B}$ and $\mathbf{S}$ is a matrix factorization problem with generalized Tikhonov regularization $trace(\mathbf{S}^T\mathbf{\Phi}\mathbf{S})$. The difference between problem (5) and matrix factorization problem with standard Tikhonov regularization $\|\mathbf{S}\|_F^2$ or sparse coding with sparsity-inducing regularization such as $\|\mathbf{S}\|_1$ is that this model tends to produce collaborative reconstruction in the sense that positively related factors attract each other and exclude negatively related factors. To justify this, one can check that if $\Phi_{i,j} > 0$, minimizing the objective function will refrain s_i and s_j to be simultaneously large or small (negative). Therefore, the latent factors learned by SLFA are not orthogonal but try to capture more deeper structures hidden in the dataset, which might be more meaningful interpretations. We validate this intuition in the experimental section using a hand crafted dataset.

The hyper-parameter ρ controls the sparsity of the $\mathbf{\Phi}$. A large ρ will result in a diagonal precision matrix $\mathbf{\Phi}$ meaning the latent factors become conditionally independent. As $\rho \to 0$, $\mathbf{\Phi}$ becomes denser. However, if we set $\rho = 0$, then the subproblem with respect to $\mathbf{\Phi}$ has a closed form solution $\mathbf{\Phi} = (\frac{1}{N}\mathbf{S}\mathbf{S}^T)^{-1}$, i.e., inverse sample covariance matrix. Plugging it back to problem (5), we have

$$\min_{\mathbf{B},\mathbf{S}} \frac{1}{N}\|\mathbf{X} - \mathbf{B}\mathbf{S}\|_F^2 + \sigma^2 \log det(\frac{1}{N}\mathbf{S}\mathbf{S}^T),$$

which doesn't have a lower bound. For unsupervised applications, we can select multiple values of ρ to obtain the $\mathbf{\Phi}$ with desired sparsity. For supervised tasks, we can use cross-validation to choose the proper value of ρ.

Relational Analysis of Latent Factors By assuming that a random vector $\mathbf{s}$ has a inverse covariance matrix $\mathbf{\Phi}$, the partial correlation between s_i and s_j can be computed by

$$\gamma_{i,j} := -\frac{\Phi_{i,j}}{\sqrt{\Phi_{i,i}}\sqrt{\Phi_{j,j}}} \tag{6}$$

Magnitude of $\gamma_{i,j}$ reflects the degree of association between s_i and s_j given all the other components $s_k, k \neq i, j$ fixed. In fact, if $\mathbf{s}$ follows a normal distribution, $\mathbf{s} \sim N(0, \mathbf{\Phi}^{-1})$, letting $\mathbf{s}_{-i}$ denote the vector $(s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_K)$, it can be shown [Yuan, 2010] that the conditional distribution of s_i given $\mathbf{s}_{-i}$ is also a normal distribution:

$$\mathbf{s}_i|\mathbf{s}_{-i} \sim N(-\mathbf{s}_{-i}^T\Phi_{ii}^{-1}\mathbf{\Phi}_{-i,i}, \Phi_{ii}^{-1}), \tag{7}$$

where $\mathbf{\Phi}_{-i,j}$ represents the the j-th column of $\mathbf{\Phi}$ with its i-th entry removed. Therefore, two variables s_i and s_j are conditionally independent given the rest if and only if $\gamma_{i,j} = \Phi_{i,j} = 0$.

Thus, we can avoid the ambiguity of the term 'relationship' between latent factors $\mathbf{B}_i$ and $\mathbf{B}_j$ by connecting it to the partial correlation between their corresponding coefficient variables. Accordingly, we propose a methodology of studying the relationship between the learned latent factors via analyzing the sparse precision matrix $\mathbf{\Phi}$ as follows.

- $\Phi_{i,j}$ is nonzero, the coefficient of factor $\mathbf{B}_j$ is predictive for the coefficient of factor $\mathbf{B}_i$.
  - $\gamma_{i,j} > 0$, s_i and s_j are positively correlated given all other $s_k, k \neq i, j$. We say $\mathbf{B}_i$ and $\mathbf{B}_j$ are positively related.
  - $\gamma_{i,j} < 0$, s_i and s_j are negatively correlated given all other $s_k, k \neq i, j$. We say $\mathbf{B}_i$ and $\mathbf{B}_j$ are negatively related.
- $\gamma_{i,j} = 0$, then s_i and s_j are conditionally independent given all other $s_k, k \neq i, j$. We say $\mathbf{B}_i$ and $\mathbf{B}_j$ have no relationship.

Relationship to Sparse Gaussian Graphical Model: We can also see SLFA as a generalization of sparse Gaussian graphical model. In fact, if the reduced dimension $K = M$, the problem (5) has trivial solution $\mathbf{B} = \mathbf{I}$ and $\mathbf{S} = \mathbf{X}$ such that the problem becomes the same as (3). When $K < M$, the subproblem with respect to $\mathbf{s}$ has solution $\mathbf{s} = (\mathbf{B}^T\mathbf{B} + \sigma^2\mathbf{\Phi})^{-1}\mathbf{x}$. Therefore, $\mathbf{s}$ can be seen as a low dimension representation under new variables which are linear combinations of original variables of $\mathbf{x}$ with weights stored in $\mathbf{W} = (\mathbf{B}^T\mathbf{B} + \sigma^2\mathbf{\Phi})^{-1}$. In this sense, SLFA results in the sparse Gaussian graphical model of $\mathbf{s} = \mathbf{W}\mathbf{x}$ and thus generalized the model from original variables to the combined variables.

3.2 An Online Algorithm

The objective function in (5) is not convex with respect to all three unknowns ($\mathbf{B}$, $\mathbf{S}$ and $\mathbf{\Phi}$) together. However, it is convex with respect to each one of $\mathbf{B}$, $\mathbf{S}$ and $\mathbf{\Phi}$ individually. Therefore, we can use Block Coordinate Descent algorithm [Bertsekas, 1999] to circularly update $\mathbf{B}$, $\mathbf{S}$ and $\mathbf{\Phi}$.

Moreover, we propose an online algorithm to tackle larger data sets, which is summarized in Algorithm 1. In the online algorithm, we randomly pick a mini-batch of observations $\mathbf{x}$ at each iteration, compute their coefficient vectors $\mathbf{s}$ and update basis matrix $\mathbf{B}$ and precision matrix $\mathbf{\Phi}$. For solving the subproblem (3), we adopt the Alternating Linearization Methods (ALM) developed by [Scheinberg et al., 2010].

Algorithm 1 Online algorithm for SLFA

```
Input: Initial guess of latent factors $\mathbf{B}$, observations $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_n]$, initial precision matrix $\mathbf{\Phi} = I$, number of iterations $T$, regularization parameters $\rho$, step-size $\gamma$ and size of mini-batch $n_b$.
  • for $t = 1$ to $T$
      – Draw a mini-batch of observations stored in $\tilde{\mathbf{X}}$ from the data set $\{\mathbf{X}_1, \ldots, \mathbf{X}_n\}$.
      – Compute the coefficient vectors $\tilde{\mathbf{S}} = (\mathbf{B}^T\mathbf{B} + \sigma^2\mathbf{\Phi})^{-1}\tilde{\mathbf{X}}$.
      – Update $\mathbf{B}$ using a gradient descent step: $\mathbf{B} \leftarrow \mathbf{B} - \frac{1}{n_b}\gamma[\mathbf{B}\tilde{\mathbf{S}} - \tilde{\mathbf{X}}]\tilde{\mathbf{S}}^T$.
      – Project all columns of $\mathbf{B}$ to the first orthant and the unit ball, i.e., $\mathbf{B} \geq \mathbf{0}$ and $\|\mathbf{B}_i\| \leq 1$.
      – Solve subproblem (3) to update the sparse inverse covariance matrix $\mathbf{\Phi}$ using a batch of recently learned coefficient vectors $\mathbf{s}$.
  • end for
```

3.3 Discriminative SLFA for Classification

Since SLFA tends to find latent factors that form a collaborative reconstruction, the K dimensional representation $\mathbf{s}$ is not the ideal feature for discriminative purpose (e.g. classification tasks). Therefore, we extend the discriminative power of SLFA by enforcing different classes to share the same latent factors but to exhibit different relational structures among latent factors. Namely, we learn the same basis matrix $\mathbf{B}$ for all C classes while for each individual class $j \in \{1, \ldots, C\}$, we learn a different precision matrix $\mathbf{\Phi}_{(j)}$. Therefore, the training process is to solve the following optimization problem:

$$
\begin{aligned}
\underset{\mathbf{B},\mathbf{S},\mathbf{\Phi}}{\text{minimize}} \; & \frac{1}{N}\|\mathbf{X}-\mathbf{BS}\|_F^2 + \sigma^2 \sum_{j=1}^{C} \{ \frac{1}{N_j} \mathrm{tr}(\mathbf{S}_{(N_j)}^T \mathbf{\Phi} \mathbf{S}_{(N_j)}) - \log det(\mathbf{\Phi}_{(j)}) + \rho \|\mathbf{\Phi}_{(j)}\|_1 \} \\
\text{subject to } & \mathbf{B} \geq \mathbf{0}, \|\mathbf{B}_k\| \leq 1, k = 1, \ldots, K, \\
& \mathbf{\Phi} \succcurlyeq 0,
\end{aligned}
$$

where $\mathbf{S}_{(N_j)}$ stores the coefficient vectors of the training data from class j. Once we learn the shared basis $\mathbf{B}$ and the precision matrices $\mathbf{\Phi}_{(j)}, j = 1, \ldots, C$, we can fit the training and testing data $\mathbf{x}$ using the model corresponding to each class to obtain the K dimensional representations $\mathbf{s}_{(j)} = (\mathbf{B}^T\mathbf{B}+\mathbf{\Phi}_{(j)})^{-1}\mathbf{B}^T\mathbf{x}$. We then have a CK dimensional representation by concatenating all the $\mathbf{s}_{(j)}$ together which can be used by standard classifier like SVM.

4 Experiment

4.1 A Toy Example

The goal of this toy experiment is to show that SLFA is able to recover more meaningful basis and their relationship than baseline latent factor models. We set up the experiment by generating 15000 images of *"bugs"*, each of which is essentially a linear combination of five latent parts shown in Figure 1 in the following way. Given 37 basis images, we first randomly select one of the five big circles as body of the 'bug'. Each shape of body is associated with four positions for legs of the bug. We then randomly pick 4 legs from its associating set of 4 small circles and 4 small strokes. However, for each leg, circle and stroke are exclusive of each other. We then combine the selected five latent parts with random coefficients that are sampled from uniform distribution and multiplied by -1 with probability 0.5. Finally, we add a randomly selected basis with small random coefficients plus Gaussian random noise to the image to introduce noise and confusion in the dataset. Few examples of data created this way are shown in Figure 1. Using SLFA and other two baseline algorithms, PCA and semi-NMF, we learn a set of latent factors and compare the result of three methods in Figures 1, we can see that only the basis generated by SLFA is very similar to the true latent factors. This is due to the fact that SLFA accounts for the partial correlation between basis in the optimization problem and encourages collaborative reconstruction.

More importantly, SLFA provides convenience of analyzing the relationship between the factors using the precision matrix Φ. In Figure 2-a, we analyze the structure learned in the precision matrix Φ. The most negatively related (exclusive) pairs (the i and j entries with highest postive entries in Φ) are circular and stroke legs which conforms fully to the generation process, since either one of them is chosen for any given location. Accordingly, the most positively related pairs are a body shape and a leg since every bug has a body and a leg.

4.2 NIPS documents

In this section, we apply SLFA to NIPS corpus[2] which contains 1740 abstracts from NIPS Conferences $1-12$ for visualization purposes. We show how SLFA is used to organize and visualize the relationship between the structured topics. SLFA is applied on the 13649

[2] http://cs.nyu.edu/ roweis/data.html

dimensional tf-idf feature vector which is normalized to have unit norm. We fix the number of topics to be 40 and tune the parameters σ and ρ to obtain Φ with proper sparsity for visualization task. In Figure 2-b, we plot a graph of topics with positive partial correlations between each other and present the first 5 keywords of a few interested topics. For example, the topic at the top is about general notions in many learning algorithms and acts as the hub point of the graph. Many of the nodes connected to the hub node contains words related to a particular learning algorithm or topic of interest. It is obvious that SLFA not only extracts underlying topic structure, but is also able to capture the correlations between topics. For example, on the far left, the topic related to cells is connected to *"motion, velocity, ..."*, *"objects, image,..."* and *"spike, neurons, ..."* nodes. This subgraph clearly represents topics related to computational vision and neuroscience. On the far right *"robot, planning, ...* node is connected to *"controller, control, ...* which represents a robotics related topic cluster.

4.3 Classification on 20 News Group

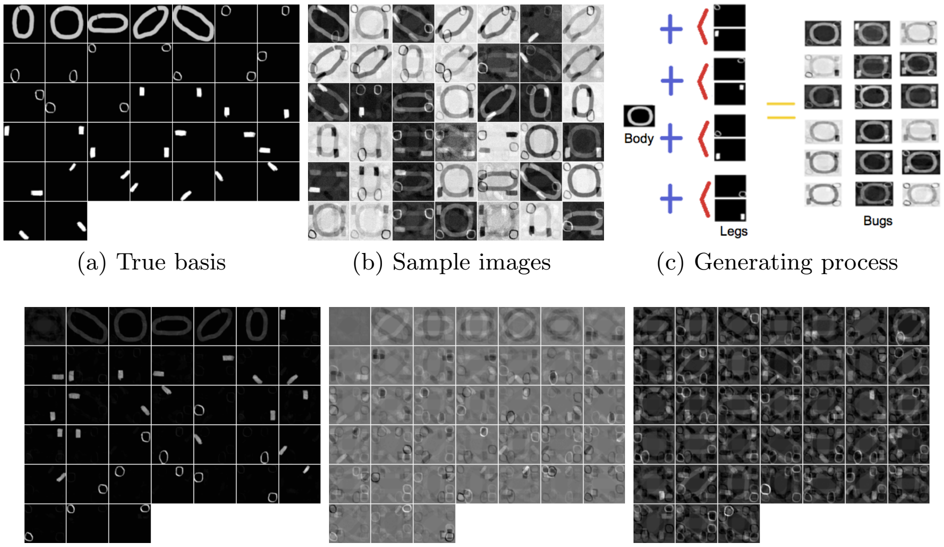
In this experiment we test the performance of SLFA model on classification of 20 News Group document data[3]. The data set consists of 18846 documents from 20 categories, which are split into training set and testing set by date. We use the frequencies of the most frequent 8000 words out of the original 26214 words as the feature of each document. We use 5 latent factor models, which are Correlated Topic Model (CTM), Latent Dirichlet Allocation (LDA), PCA, our proposed model SLFA and its extension DiscSLFA, to find $K = 36, 49, 64$ and 64 latent topics and then perform LibSVM [Fan et al., 2008] to train a linear classifier on the low dimensional representations. td-idf transformation is used for PCA, SLFA and DiscSLFA and 5-fold cross-validation on the training set is used to tune the parameters in SLFA and DiscSLFA. We report the average accuracy of 20 categories in Figure 2-c and show that SLFA performs slightly better than classical dimension reduction algorithm PCA which produces orthogonal representations. More importantly DiscSFLA, which uses supervised label information to learn separate precision matrices per class, significantly improves the accuracy.

4.4 Gene Microarray Data

We test our model for the classification task on a breast cancer microarray data set obtained from [Jacob et al., 2009]. SLFA could deeply explore the latent information in this data set and can even compete with state-of-the-art classification methods which utilized extra biological evidence. This data set contains gene expression values of $8,141$ genes for 295 breast cancer tumor samples (with 78 metastatic and 217 non-metastatic). We compare six methods on their classification error rates, which include Lasso [Tibshirani, 1996], GLasso [Jacob et al., 2009], Linear SVM classifier [Fan et al., 2008], PCA with SVM classifier, SLFA with SVM classifier and DiscSLFA with SVM classifier. For GLasso (i.e. a logistic regression approach using the graph-guided sparsity), a prior biological network information is provided ($42,594$ known edges between genes) to construct the graphical regularization.

Since the sample size is very small, we run 10-fold cross validation and use the averaged error rate on the validation set to indicate the predictive performance of different methods. The test is repeated for 50 times and for each time all methods use the same spit of training and validation sets. The boxplot of the CV error rate is shown in Figure 2-d. We can observe that SLFA and DiscSLFA have lower error rate than methods such as LASSO, SVM and PCA. Furthermore, compared to the method GLasso [Jacob et al., 2009] which construct regularization from external information, our method based on SLFA is even better, which indicates SLFA can extract deep structural information hidden in the data. Unlike document data using 20 News Group, DiscSLFA doesn't perform superior than SLFA. This most likely due to the fact that the sample size for each class is to small and DiscSLFA ends up over-exploring the information in training data.

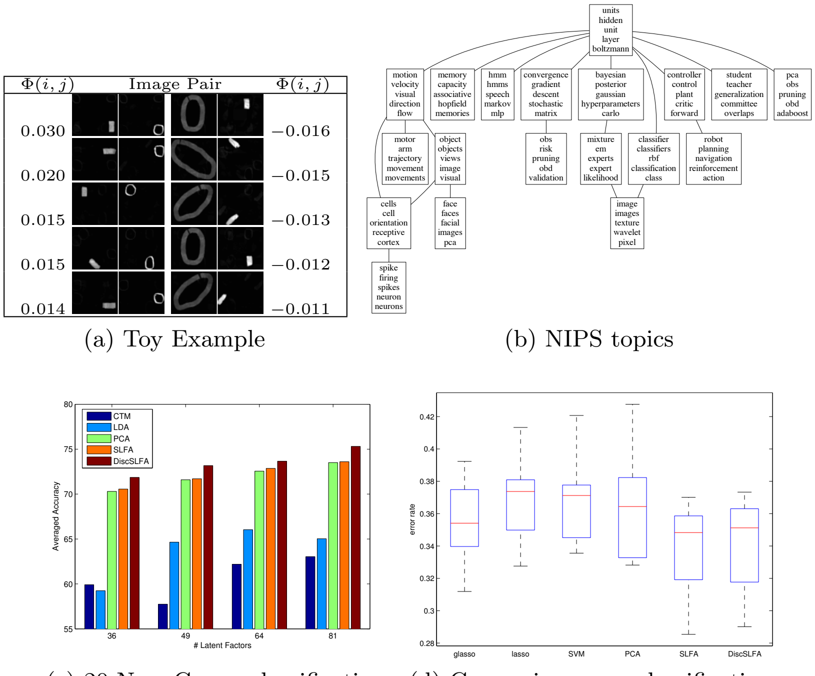
[3] http://people.csail.mit.edu/jrennie/20Newsgroups/



(a) True basis (b) Sample images (c) Generating process

(d) SLFA (e) PCA (f) Semi-NMF

Figure 1: Upper figures show the images of true basis, generated sample images and one example of generating process. Lower figures show the basis recovered by SLFA, PCA and semi-NMF. The basis learned by PCA and semi-NMF do not reveal any underlying structure of generating process.



(a) Toy Example (b) NIPS topics

(c) 20 NewsGroup classification (d) Gene microarray classification

Figure 2: Experiment result: (a) The table shows the five largest and and five smallest entries in Φ and their corresponding B_i and B_j pairs. For $\Phi(i,j) > 0$, i and j are negatively related (exclusive), for $\Phi(i,j) < 0$, i and j are positively related (supportive). (b) Positively related topics discovered from NIPS text corpus. (c)Classification performance of different methods on 20 News Group Data. DiscSLFA improves classification rate significantly. (d)Cross-validation error rate by different methods on Gene Micro-array data. Both SLFA and DiscSLFA perform better than other methods.

References

[Bertsekas, 1999] Bertsekas, D. (1999). *Nonlinear programming.* Athena Scientific Belmont, MA.

[Chen et al., 2011] Chen, X., Qi, Y., Bai, B., Lin, Q., and Carbonell, J. (2011). Sparse latent semantic analysis. In *SIAM International Conference on Data Mining (SDM).*

[Deerwester et al., 1990] Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science.*

[Fan et al., 2008] Fan, R., Chang, K., Hsieh, C., Wang, X., and Lin, C. (2008). Liblinear: A library for large linear classification. *The Journal of Machine Learning Research.*

[Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning.* Springer.

[Jacob et al., 2009] Jacob, L., Obozinski, G., and Vert, J. (2009). Group lasso with overlap and graph lasso. In *Proceedings of the 26th Annual ICML*. ACM.

[Jordan, 1998] Jordan, M. (1998). *Learning in graphical models.* Kluwer Academic Publishers.

[Lee and Seung, 1999] Lee, D. and Seung, H. S. (1999). Algorithms for non-negative matrix factorization. In *NIPS.*

[Olshausen et al., 1996] Olshausen, B. et al. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature.*

[Scheinberg et al., 2010] Scheinberg, K., Ma, S., and Goldfarb, D. (2010). Sparse inverse covariance selection via alternating linearization methods. *NIPS.*

[Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological).*

[Yuan, 2010] Yuan, M. (2010). High dimensional inverse covariance matrix estimation via linear programming. *The Journal of Machine Learning Research*, 11:2261–2286.

[Yuan and Lin, 2007] Yuan, M. and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika.*