Semi-supervised Low-Rank Mapping Learning for Multi-label Classification

Liping Jing, Liu Yang, Jian Yu
Beijing Key Lab of Traffic Data Analysis and Mining
Beijing Jiaotong University
{lpjing, 11112091, jianyu}@bjtu.edu.cn

Michael K. Ng
Department of Mathematics
Hong Kong Baptist Unviersity
mng@math.hkbu.edu.hk

Abstract

Multi-label problems arise in various domains including automatic multimedia data categorization, and have generated significant interest in computer vision and machine learning community. However, existing methods do not adequately address two key challenges: exploiting correlations between labels and making up for the lack of labeled data or even missing labels. In this paper, we proposed a semi-supervised low-rank mapping (SLRM) model to handle these two challenges. SLRM model takes advantage of the nuclear norm regularization on mapping to effectively capture the label correlations. Meanwhile, it introduces manifold regularizer on mapping to capture the intrinsic structure among data, which provides a good way to reduce the required labeled data with improving the classification performance. Furthermore, we designed an efficient algorithm to solve SLRM model based on alternating direction method of multipliers and thus it can efficiently deal with large-scale datasets. Experiments on four real-world multimedia datasets demonstrate that the proposed method can exploit the label correlations and obtain promising and better label prediction results than state-of-the-art methods.

1. Introduction

With the rapid growth of online content such as images, videos, web pages, it is crucial to design a scalable and effective classification system to automatically organize, store, and search the content. In conventional classification, each instance is assumed to belong to exactly one class among a finite number of candidate classes. However, in modern applications, an instance can have multiple labels. For example, an image can be annotated by many conceptual tags in semantic scene classification. Multi-label data have ubiquitously occurred in many application domains: multimedia information retrieval, tag recommendation, query categorization, gene function prediction, medical diagnosis, drug discovery and marketing. An important and challenging research problem [10, 33] in multi-label learning is how to exploit and make use of label correlations.

In literatures, three label-embedding strategies [1] are usually adopted to identify label correlations. The first is derived from side information [4] such as text descriptions and taxonomies. The second is data-independent, i.e., it explicitly and implicitly exploits multi-label relationships from only label information. The main idea of explicit methods is to extract label relationships, and then construct label hierarchies [24], co-occurrence pairs [12], hypergraphs [26], networks [9], label power-sets [28] and etc. These methods can be quite effective in multi-label learning, but their computational complexities are usually high especially when the number of labels is large. In implicit methods, a low-dimensional latent space is learned to represent the original data information. Hsu et al. [14] adopted compressed sensing technique and Tai and Lin [27] took advantage of the principal components analysis to generate the low-dimensional latent space. However, these data-independent methods only take into account the label information.

The last is learned embedding which uses both feature and label information. Hariharan et al. [13] proposed a max-margin multi-label model to do correlated prediction among labels. Zhang and Schneider [34] tried to find proper multi-label output codes with the aid of canonical correlation analysis. Huang and Zhou [15] identified the local label correlations by separating data objects into different groups. Chen and Lin [8] extended the method [27] by integrating feature information. Lin et al. [18] aimed to learn a low-dimensional latent space to represent the given data. Recall that multi-label classification aims to learn a mapping from a feature space to a label space, thus, both feature information and label information should be used to exploit the label correlations. However, these existing methods may be time-consuming [13, 15] or mostly depends on the predefined latent space size [34, 8, 18].

On the other hand, supervised learning methods train classifiers based on sufficiently enough multi-labeled instances. However, it is very expensive to obtain multi-

labeled instances. In practice, we are always faced with a small number of multi-labeled instances and a large amount of unlabeled instances. Therefore, it is very important to develop semi-supervised multi-label learning methods that can use both multi-labeled data and unlabeled data together to deal with this important problem. Li and Guo [16] and Vasisht et al. [29] integrated active learning techniques into multi-label learning process to improve the prediction performance. Cabral et al. [5] constructed a feature-plus-label joint matrix by integrating both multi-labeled and unlabeled data together. The main step of their iterative algorithm is to compute the singular value decomposition of the joint matrix, which is computational expensive when the number of objects, the number of features and the number of labels are large.

The main contribution of this paper is to develop a novel method for multi-label learning when there is only a small number of multi-labeled data. Our main idea is to design a Semi-supervised Low-Rank Mapping (SLRM) from a feature space to a label space based on given multi-label data. The low-rank based regularization of the mapping can effectively exploit the label and feature correlations because the label/feature component vectors can be explicitly described via the left/right singular vectors of the mapping. In order to make use of multi-labeled and unlabeled data, we also construct the mapping based on manifold regularization. In the regularization, we enforce that when two instances are close in the feature space, their new representation based on mapping should be close. In this case, the mapping is able to capture the intrinsic geometric structure among instances in both feature space and label space. As a virtuous by-product, SLRM can handle missing labels because it has ability to fill such missing entries with label correlations and intrinsic structure among data, which is crucial as we may not have access to all the true labels of each training instance in most real applications [30].

An algorithm for finding SLRM mapping is developed by using the alternating direction method of multipliers which can handle large datasets very efficiently. In the paper, we have conducted extensive experiments on four multi-label multimedia datasets. The reported results demonstrate the effectiveness of the proposed SLRM method. In particular, SLRM outputs promising and better multi-label classification performance than the other testing methods (CPLST [8], FAIE [18], MLLOC [15], MC [5], and MIML [29]).We also illustrate that the proposed algorithm is very efficient for large-scale datasets.

The rest of this paper is organized as follows. In Section 2, we present the proposed SLRM model. In Section 3, we develop the algorithm for finding the mapping. In Section 4, we report the experimental results and compare the performance of different methods. Some concluding remarks are given in Section 5.

2. Semi-supervised Low-Rank Mapping

2.1. Notations

Throughout the paper, matrices are denoted by uppercase bolded letters (e.g., $\mathbf{A}$), vectors are denoted by lowercase bolded letters (e.g., $\mathbf{a}$), and scalars appear as lowercase letters (e.g., a). The ith column of $\mathbf{A}$ is denoted as $\mathbf{a}_i$. The set of real numbers is denoted as $\mathbb{R}$. A variety of norms on real-valued vectors and matrices will be used. For example, $\|\mathbf{A}\|_F = \sqrt{\sum_i \sum_j a_{i,j}^2} = \sqrt{tr(\mathbf{A}^T\mathbf{A})}$ is the Frobenius norm, where $tr(\cdot)$ denotes the trace of a matrix, and $\mathbf{A}^T$ is the transpose of $\mathbf{A}$. The nuclear norm, $\|\mathbf{A}\|_*$, is defined as the sum of singular values, i.e., $\|\mathbf{A}\|_* = \sum_i \sigma_i(\mathbf{A})$ where $\sigma_i(\mathbf{A})$ is the ith singular value of matrix $\mathbf{A}$.

Given a set of labeled data with n_l instances $\{(\hat{\boldsymbol{x}}_i, \boldsymbol{y}_i)\}_{i=1}^{n_l}$, where $\hat{\boldsymbol{x}}_i \in \mathbb{R}^d$ and $\boldsymbol{y}_i \in \mathbb{R}^k$ are respectively the d-dimensional feature vector and k-dimensional label vector of the ith labeled data, the traditional multi-label learning aims to find a mapping function from $\hat{\mathbf{X}} = [\hat{\boldsymbol{x}}_1\ \hat{\boldsymbol{x}}_2\ \cdots\ \hat{\boldsymbol{x}}_{n_l}]$ to $\mathbf{Y} = [\boldsymbol{y}_1\ \boldsymbol{y}_2\ \cdots\ \boldsymbol{y}_{n_l}]$ using labeled data only. Each entry of the label vector indicates whether the current instance belongs to the corresponding class. In real applications, there are amounts of unlabeled data with n_u instances denoted as $\check{\mathbf{X}} = [\check{\boldsymbol{x}}_1\ \check{\boldsymbol{x}}_2\ \cdots\ \check{\boldsymbol{x}}_{n_u}]$, where $\check{\boldsymbol{x}}_i \in \mathbb{R}^d$. The whole dataset is denoted as $\mathbf{X} = [\hat{\mathbf{X}}, \check{\mathbf{X}}]$ with n instances and $n = n_l + n_u$. Our goal is to effectively and efficiently find a good mapping from $\hat{\mathbf{X}}$ to $\mathbf{Y}$ using the whole dataset $\mathbf{X}$.

2.2. The Proposed Model

The proposed semi-supervised multi-label learning model can be stated in the following framework:

$$\min_{\mathtt{f}} \sum_{i=1}^{n_l} \mathcal{L}(\hat{\mathbf{x}}_i, \mathbf{y}_i, \mathtt{f}) + \lambda\Phi(\mathtt{f}) + \gamma\Psi(\mathtt{f}) \qquad (1)$$

where $\mathcal{L}(\cdot)$ is the loss function measuring the labeling approximation error between the given multi-labeled data and the prediction result, $\Phi(\mathtt{f})$ is the regularization to control the complexity of $\mathtt{f}$ and $\Psi(\mathtt{f})$ is the regularization to control the smoothness of $\mathtt{f}$ by requiring that two instances are close in the label space when they are close in the feature space. Moreover, λ and γ are two parameters to control the balance among the three terms in the objective function.

2.3. The Data Fitting Term

The choice of loss function $\mathcal{L}(\cdot)$ usually depends on application domains. Three convex functions are usually used such as the hinge loss, logistic loss and least square loss functions. Among them, the least square loss function is always adopted for classification problems. As shown in [11], the least square loss function can provide comparable performance to the hinge loss function. Moreover, it

has shown that the least squares loss function is universally Fisher consistent and shares the same population minimizer with the squared hinge loss function [36]. Thus, we formulate a tractable optimization problem by using the least square loss function as follows:

$$\mathcal{L}(\hat{\mathbf{x}}_i, \mathbf{y}_i, \mathtt{f}) = \|\mathtt{f}(\hat{\mathbf{x}}_i) - \mathbf{y}_i\|_2^2 \tag{2}$$

More precisely, we express the mapping as a linear transformation $\mathbf{U} \in \mathbb{R}^{k\times d}$ from the feature space $\mathbb{R}^d$ to the label space $\mathbb{R}^k$:

$$\mathcal{L}(\hat{\mathbf{x}}_i, \mathbf{y}_i, \mathtt{f}) = \|\mathbf{U}\hat{\mathbf{x}}_i - \mathbf{y}_i\|_2^2. \tag{3}$$

2.4. The Regularization of Complexity

The linear transformation $\mathbf{U}$ can be characterized by its singular value decomposition:

$$\sum_{j=1}^{r} \mathbf{p}_j(\mathbf{U})\sigma_j(\mathbf{U})\big(\mathbf{q}_j(\mathbf{U})\big)^T \tag{4}$$

where $r = \min\{k, d\}$, $p_j(\mathbf{U}) \in \mathbb{R}^k$ and $q_j(\mathbf{U}) \in \mathbb{R}^d$ are singular vectors of $\mathbf{U}$, and $\sigma_j(\mathbf{U})$ is the jth singular value of $\mathbf{U}$. It is interesting to note that all the singular values are real and non-negative. With loss of generality, we assume that $\sigma_1(\mathbf{U}) \geq \sigma_2(\mathbf{U}) \geq \cdots \geq \sigma_r(\mathbf{U})$. Therefore, the complexity of $\mathbf{U}$ can be measured by the sum of its singular values, i.e., how many singular values of $\mathbf{U}$ or singular vectors $\mathbf{p}_j(\mathbf{U})$ and $\mathbf{q}_j(\mathbf{U})$ we should keep. When the largest r' singular values of $\mathbf{U}$ are kept, the corresponding rank of $\mathbf{U}$ is also equal to r'. Equivalently, the nuclear norm regularization is employed to measure the complexity of $\mathbf{U}$:

$$\Phi(\mathbf{U}) = \|\mathbf{U}\|_* = \sum_{j=1}^{r} \sigma_j(\mathbf{U}). \tag{5}$$

We remark that the minimization of nuclear norm is a convex optimization problem. In literatures, this regularization has been applied to many applications such as dimension reduction [31], multi-task learning [2], subspace structure identification [20], multi-label learning [5] and etc.

Moreover, based on the minimization of nuclear norm, the linear transformation of each data point $\hat{\mathbf{x}}_i$ can be given by

$$\mathbf{U}\hat{\mathbf{x}}_i = \sum_{j=1}^{r'} \sigma_j(\mathbf{U})[\big(\mathbf{q}_j(\mathbf{U})\big)^T\hat{\mathbf{x}}_i]\mathbf{p}_j(\mathbf{U}). \tag{6}$$

We note that this resulting vector is in the label space, and it is a linear combination of label-component vectors: $\mathbf{p}_1(\mathbf{U}), \mathbf{p}_2(\mathbf{U}), \cdots, \mathbf{p}_{r'}(\mathbf{U})$ which correspond to the largest r' singular values in the singular value decomposition of $\mathbf{U}$. Therefore, the label correlations can be recognized and represented by these label-component vectors. Meanwhile, we expect that even when there are some missing labels and the number of labeled training data is small, the regularization term can help to identify correct labels based on these label-component vectors.

In the data-fitting term (3), we search for a linear combination of the label-component vectors $\{\mathbf{p}_j(\mathbf{U})\}_{j=1}^{r'}$ such that it is close to $\mathbf{y}_i$. The optimal coefficient in the linear combination contains two parts: one is the magnitude of the singular value $\sigma_j(\mathbf{U})$, the other is transformed feature value $\big(\mathbf{q}_j(\mathbf{U})\big)^T\hat{\mathbf{x}}_i$ which is determined by r' feature-component vectors $\{\mathbf{q}_j(\mathbf{U})\}_{j=1}^{r'}$. By using these feature-component vectors and label-component vectors, the feature correlations and label correlations can be recognized.

2.5. The Regularization of Smoothness

For label prediction, the main purpose of $\mathbf{U}$ is to propagate the semantic information from feature space to label space. Therefore, an instance $\mathbf{x}$ (as a feature vector) can be labeled via its corresponding label vector $\mathbf{y} = \mathbf{U}\mathbf{x}$. It is intuitive that the predicted label vectors should have ability to keep the intrinsic structure among data feature vectors. In other words, if two data points $\mathbf{x}_i$ and $\mathbf{x}_j$ are close in the intrinsic geometry of the data distribution, then the predicted label vectors $\mathbf{y}_i$ (i.e., $\mathbf{U}\mathbf{x}_i$) and $\mathbf{y}_j$ (i.e., $\mathbf{U}\mathbf{x}_j$) are also close to each other. This is referred to local invariance assumption [3], which is well studied in manifold learning theory. It plays an essential role in the development of various kinds of algorithms including semi-supervised learning algorithms [35] and matrix factorizations [6].

To model the intrinsic geometric structure among data, a nearest neighbor graph is usually constructed with n vertices (each vertex corresponding to one instance). For each instance, its c nearest neighbors are selected according to the similarity between instances, an edge is assigned between the instance and its neighbors. Typically, the heat kernel weight with self-tuning technique (for parameter σ) [32] is adopted here as the edge weight if two points are connected $a_{i,j} = \exp\left(\frac{-\|\mathbf{x}_i-\mathbf{x}_j\|^2}{\sigma}\right)$, otherwise $a_{i,j} = 0$. Then, we can generate an edge matrix $\mathbf{A} = [a_{i,j}]$ corresponding to the c-nearest neighbor graph. Based on matrix $\mathbf{A}$, the local invariance assumption can be formulated via a manifold regularization

$$\Psi(\mathbf{U}) = \frac{1}{2}\sum_{i,j=1}^{n} a_{i,j}\,\|\mathbf{U}\mathbf{x}_i - \mathbf{U}\mathbf{x}_j\|_2^2 = tr((\mathbf{U}\mathbf{X})\mathbf{L}(\mathbf{U}\mathbf{X})^T), \tag{7}$$

where $\mathbf{L}$ is graph Laplacian of matrix $\mathbf{A}$ defined as $\mathbf{L} = \mathbf{D} - \mathbf{A}$, and $\mathbf{D}$ is a diagonal matrix whose main diagonal entries are column sums of $\mathbf{A}$, i.e., $d_{i,i} = \sum_{i=1}^{n} a_{i,j}$. In this setting, both labeled and unlabeled data $\mathbf{X} = [\hat{\mathbf{X}}, \check{\mathbf{X}}]$ can be used to construct $\mathbf{L}$.

By combining data fitting term in (3), the regularization

of complexity in (5) and the regularization of smoothness in (7), the resulting **S**emi-supervised **L**ow-**R**ank **M**apping model for multi-label learning (SLRM) can be developed:

$$\min_{\mathbf{U}} ||\mathbf{U}\hat{\mathbf{X}} - \mathbf{Y}||_F^2 + \lambda||\mathbf{U}||_* + \gamma tr((\mathbf{U}\mathbf{X})\mathbf{L}(\mathbf{U}\mathbf{X})^T). \quad (8)$$

We remark that the resulting model is a convex optimization problem as each term in the objective function is convex with respect to $\mathbf{U}$. The SLRM model is able to learn a mapping function $\mathbf{U}$ that can capture the correlations among labels and the geometric structure among data in the feature space. This property will make SLRM useful in handling multi-label classification with a few labeled data and missing labels.

3. The Proposed Algorithm

The optimization problem (8) is convex and can be solved by various methods. In this paper, we employ the alternating direction method of multipliers [17] method to find the optimal solution. By introducing an auxiliary variable $\mathbf{V} \in \mathbb{R}^{k\times d}$, we can convert (8) to the following equivalent problem:

$$\begin{aligned}\min_{\mathbf{U},\mathbf{V}} \quad & \tfrac{1}{2}\|\mathbf{U}\hat{\mathbf{X}} - \mathbf{Y}\|_F^2 + \lambda\|\mathbf{V}\|_* + \frac{\gamma}{2} tr((\mathbf{U}\mathbf{X})\mathbf{L}(\mathbf{U}\mathbf{X})^T) \\ & \text{subject to } \mathbf{U} = \mathbf{V}. \end{aligned} \quad (9)$$

The augmented Lagrange function of (9) is given by

$$\begin{aligned}\min_{\mathbf{U},\mathbf{V},\boldsymbol{\Upsilon}} & \frac{1}{2}\|\mathbf{U}\hat{\mathbf{X}} - \mathbf{Y}\|_F^2 + \frac{\gamma}{2} tr((\mathbf{U}\mathbf{X})\mathbf{L}(\mathbf{U}\mathbf{X})^T) + \\ & \frac{\beta}{2}||\mathbf{U} - \mathbf{V}||_F^2 + tr(\boldsymbol{\Upsilon}^T(\mathbf{U} - \mathbf{V})) + \lambda||\mathbf{V}||_* \end{aligned} \quad (10)$$

where $\boldsymbol{\Upsilon} \in \mathbb{R}^{k\times d}$ is the Lagrange multipliers, and β is a positive number which can be adaptively updated [19]. Now the above optimization problem can be minimized with respect to each variable ($\mathbf{U}$, $\mathbf{V}$, and $\boldsymbol{\Upsilon}$) by fixing the other two variables in an alternating manner.

3.1. The Update of $\mathbf{U}$

When we fix $\mathbf{V} = \mathbf{V}^{(m)}$ and $\boldsymbol{\Upsilon} = \boldsymbol{\Upsilon}^{(m)}$, $\mathbf{U}$ can be determined by solving the following problem:

$$\begin{aligned}\min_{\mathbf{U}} & \frac{1}{2}||\mathbf{U}\hat{X} - \mathbf{Y}||_F^2 + \frac{\gamma}{2} tr((\mathbf{U}\mathbf{X})\mathbf{L}(\mathbf{U}\mathbf{X})^T) + \\ & \frac{\beta}{2}||\mathbf{U} - \mathbf{V}^{(m)}||_F^2 + tr((\boldsymbol{\Upsilon}^{(m)})^T(\mathbf{U} - \mathbf{V}^{(m)})) \end{aligned} \quad (11)$$

Note that $\cdot^{(m)}$ refers at the mth iteration index. According to (11), $\mathbf{U}^{(m+1)}$ can be updated by solving a linear system:

$$\mathbf{U}\hat{\mathbf{X}}\hat{\mathbf{X}}^T - \mathbf{Y}\hat{\mathbf{X}}^T + \gamma\mathbf{U}\mathbf{X}\mathbf{L}\mathbf{X}^T + \beta\mathbf{U} - \beta\mathbf{V}^{(m)} + \boldsymbol{\Upsilon}^{(m)} = 0.$$

The closed form solution is given as follows:

$$\mathbf{U}^{(m+1)} := (\hat{\mathbf{X}}\hat{\mathbf{X}}^T + \gamma\mathbf{X}\mathbf{L}\mathbf{X}^T + \beta\mathbf{I}_d)^{-1}(\mathbf{Y}\hat{\mathbf{X}}^T + \beta\mathbf{V}^{(m)} - \boldsymbol{\Upsilon}^{(m)}). \quad (12)$$

It is clear the inverse of $(\hat{\mathbf{X}}\hat{\mathbf{X}}^T + \gamma\mathbf{X}\mathbf{L}\mathbf{X}^T + \beta\mathbf{I}_d)$ exists as the matrix is symmetric positive definite.

3.2. The Update of $\mathbf{V}$

Similarly, when we fix $\mathbf{U} = \mathbf{U}^{(m+1)}$ and $\boldsymbol{\Upsilon} = \boldsymbol{\Upsilon}^{(m)}$, $\mathbf{V}$ can be determined by solving the following problem:

$$\min_{\mathbf{V}} \frac{\beta}{2}||\mathbf{U}^{(m+1)} - \mathbf{V}||_F^2 + tr((\boldsymbol{\Upsilon}^{(m)})^T(\mathbf{U}^{(m+1)} - \mathbf{V})) + \lambda||\mathbf{V}||_*. \quad (13)$$

The optimization problem can be further reduced to the following form:

$$\min_{\mathbf{V}} \frac{1}{2}\left\|\mathbf{U}^{(m+1)} + \frac{\boldsymbol{\Upsilon}^{(m)}}{\beta} - \mathbf{V}\right\|_F^2 + \frac{\lambda}{\beta}||\mathbf{V}||_*. \quad (14)$$

It is obvious that this optimization problem is convex and has a unique minimizer. The solution $\mathbf{V}^{(m+1)}$ can be solved via the singular value thresholding operator [7]:

$$\mathbf{V}^{(m+1)} := \mathcal{D}_{\frac{\lambda}{\beta}}\left[\mathbf{U}^{(m+1)} + \frac{\boldsymbol{\Upsilon}^{(m)}}{\beta}\right], \quad (15)$$

where $\mathcal{D}_{\frac{\lambda}{\beta}}[\cdot]$ is the output matrix given by the singular vectors of the input matrix and its modified singular values are shrinkaged by the formula:

$$\begin{aligned} & sgn\left(\sigma_i\left(\mathbf{U}^{(m+1)} + \frac{\boldsymbol{\Upsilon}^{(m)}}{\beta}\right)\right) \times \\ & \max\left\{\left|\mathbf{U}^{(m+1)} + \frac{\boldsymbol{\Upsilon}^{(m)}}{\beta}\right| - \frac{\lambda}{\beta}, 0\right\}. \end{aligned} \quad (16)$$

3.3. The Update of $\boldsymbol{\Upsilon}$

Once having $\mathbf{U} = \mathbf{U}^{(m+1)}$ and $\mathbf{V} = \mathbf{V}^{(m+1)}$, we determine $\boldsymbol{\Upsilon}$ by considering the amount of violation of the constraint $\mathbf{U}^{(m+1)} = \mathbf{V}^{(m+1)}$. This amount will be used to update $\boldsymbol{\Upsilon}^{(m+1)}$ via

$$\boldsymbol{\Upsilon}^{(m+1)} := \boldsymbol{\Upsilon}^{(m)} + \beta(\mathbf{U}^{(m+1)} - \mathbf{V}^{(m+1)}). \quad (17)$$

3.4. Computational Complexity

For each iteration, updating U in (12) requires the construction of $(\hat{\mathbf{X}}\hat{\mathbf{X}}^T + \gamma\mathbf{X}\mathbf{L}\mathbf{X}^T + \beta\mathbf{I}_d)$ and $(\mathbf{Y}\hat{\mathbf{X}}^T + \beta\mathbf{V}^{(m)} - \boldsymbol{\Upsilon}^{(m)})$, which will cost $O(d^2n_l + dn^2 + kdn_l)$. The inverse of $(\hat{\mathbf{X}}\hat{\mathbf{X}}^T + \gamma\mathbf{X}\mathbf{L}\mathbf{X}^T + \beta\mathbf{I}_d)$ with $O(d^3)$ complexity is not necessary to be computed at each iteration. The dominant cost for updating $\mathbf{V}$ in (15) is the computation of the singular value thresholding operator, and its complexity is $\min\{O(dk^2), O(d^2k)\}$. Updating $\boldsymbol{\Upsilon}$ in (17) only needs matrix plus or minus operation and it costs $O(dk)$. Consequently, with alternating direction of multipliers method, the complexity of solving SLRM is $O(dk^2 + d^2n_l + dn^2 + d^3)$. We remark that the computational complexity can be reduced when we consider inexact version of alternating direction method of multipliers [21] or the suitable surrogate functions [23].

3.5. The Related Work

According to Lemma 1 in [25], the nuclear norm of matrix $\mathbf{U} \in R^{k\times d}$ is equal to identifying the maximum margin matrix factorization via

$$\|\mathbf{U}\|_* = \arg\min_{\mathbf{U}=\mathbf{P}\mathbf{W}^T} \frac{1}{2}\big(\|\mathbf{W}\|_F^2 + \|\mathbf{P}\|_F^2\big), \qquad (18)$$

where $\mathbf{W} \in \mathbb{R}^{d\times b}$ and $\mathbf{P} \in \mathbb{R}^{k\times b}$ are required matrices. For multi-label learning problems, $\mathbf{W}$ can be taken as the new representation of features in the latent space $\mathbb{R}^b$, where each row vector of $\mathbf{W}$ indicates a feature. $\mathbf{P}$ gives the new representation of labels in the latent space $\mathbb{R}^b$. Each row vector of $\mathbf{P}$ represents a label. The model has ability to capture the intrinsic information from both feature space and label space. In order to identify the latent information in label space, Tai and Lin [27] took the original label space as a hypercube and mined its principal components by

$$\max_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{Y}\mathbf{Y}^T\mathbf{P}) \text{ s.t. } \mathbf{P}^T\mathbf{P} = \mathbf{I}, \qquad (19)$$

where $\mathbf{P} \in \mathbb{R}^{k\times b}$ consists of the normalized eigenvectors of $\mathbf{Y}\mathbf{Y}^T$ corresponding to its b largest eigenvalues. This method is named as PLST. In [8], Chen and Lin extended it to the CPLST model by integrating the labeled data information $\hat{\mathbf{X}}$ via

$$\max_{\mathbf{P}} tr(\mathbf{P}^T\mathbf{Y}\hat{\mathbf{X}}^\dagger\hat{\mathbf{X}}\mathbf{Y}^T\mathbf{P}) \text{ s.t. } \mathbf{P}^T\mathbf{P} = \mathbf{I}, \qquad (20)$$

where $\hat{\mathbf{X}}^\dagger$ is the pseudo-inverse of $\hat{\mathbf{X}}$. Similarly, $\mathbf{P} \in \mathbb{R}^{k\times b}$ indicates the principal components of the labeled data.

Recently, Lin et al. [18] proposed an implicit label space encoding method (FAIE), which jointly maximizes the recoverability of the label space and the predictability of the feature space via

$$\max_{\mathbf{C}} tr(\mathbf{C}^T(\mathbf{Y}^T\mathbf{Y} + \alpha\hat{\mathbf{X}}^T(\hat{\mathbf{X}}\hat{\mathbf{X}}^T)^{-1}\hat{\mathbf{X}})\mathbf{C}) \text{ s.t. } \mathbf{C}^T\mathbf{C} = \mathbf{I}, \qquad (21)$$

where $\mathbf{C} \in \mathbb{R}^{n_l\times b}$ indicates the relationships between data instances and the latent space. We note that $\mathbf{C}$ cannot explicitly reflect the correlation between labels which is a main point in multi-label learning. Based on the proposed SLRM model and (18), $\mathbf{C}$ can be easily recovered via a linear transformation $\hat{\mathbf{X}}^T\mathbf{W}$. We remark that these three methods (PLST, CPLST, FAIE) have to predefine the size of latent space (b) appropriately.

4. Experimental Results and Discussion

4.1. Datasets

We evaluated the performance of our method for multi-label classification on four datasets including *MSRC*[1], *SUN*attribute database [22] and two Mulan multimedia datasets[2] (*Core5K* and *Mediamill*). Among them, *MSRC* images are represented via bag of words on sampled patches. *SUN* [3] images are represented via low-level features (gist descriptors). For *Core5K* and *Mediamill*, the preprocessed data are directly downloaded from Mulan website. Since we wish to study the mechanisms of multi-label classification model in this work, we stuck with the feature set for each dataset through the experimental procedure once it is chosen.

More detailed information about the data size can be found in Table 1, where n is the number of instances, d is the number of features, k is the number of labels/classes. The *cardinality* column is defined as the average number of labels per instance. In these datasets, the number of labels varies from 23 to 374, the *cardinality* varies from 2.508 to 15.526, and the number of instances in different classes changes in a large range (e.g. the class size in *SUN* varies from 141 to 11878), thus it is a challenging task to predict the label information for such multi-label datasets.

Dataset	Domain	n	d	k	*cardinality*
MSRC	image	591	512	23	2.508
Corel5K	image	5000	499	374	3.522
SUN	image	14240	512	102	15.526
Mediamill	video	43907	210	101	4.376

Table 1. Multi-label dataset summary.

4.2. Methodology

In order to demonstrate the performance of the proposed SLRM method, we take CPLST [8], FAIE [18], MLLOC [15], MC [5], and MIML [29] in our comparison. We do not include PLST [27] because earlier work [8] has shown that they are inferior to CPLST. We run MLLOC[4] with the Matlab codes provided by the authors, and the other five methods are implemented in Matlab (All are run on Windows with 4G memory and 2Ghz CPU). Among them, the first three methods train the classifiers only using the labeled dataset. Like SLRM, the later two methods (the semi-supervised MC and active learning method MIML), take advantage of both labeled and unlabeled datasets.

For CPLST and FAIE, the number of reduced latent space dimensions (b in Section 3.5) is selected from $\{\lfloor 0.05k\rfloor, \lfloor 0.1k\rfloor, \lfloor 0.2k\rfloor, \lfloor 0.3k\rfloor, \lfloor 0.4k\rfloor, \lfloor 0.5k\rfloor\}$ if k is greater than 10, otherwise b is tuned in range $[2, k]$ with each step increment by 1. In the learning stage, both CPLST and FAIE are coupled with linear regression for label prediction. The regularization parameter in MLLOC and MC, and the trade-off parameter in FAIE, λ and γ in SLRM are

[1] http://research.microsoft.com/en-us/projects/ObjectClassRecognition/

[2] http://mulan.sourceforge.net/datasets-mlc.html

[3] https://cs.brown.edu/ gen/sunattributes.html

[4] http://lamda.nju.edu.cn/code_MLLOC.ashx

Dataset	Evaluation	CPLST	FAIE	MLLOC	MC	MIML	SLRM
MSRC	AUC	0.7887	0.7780	0.5400	0.7857	0.8133	**0.8253**
	Macro-F1	0.3317	0.3467	0.1048	0.2541	0.4083	**0.4481**
	Micro-F1	0.5109	0.5357	0.3692	0.4196	0.5538	**0.5890**
	Accuracy	0.3281	0.3344	0.2070	0.2353	0.2801	**0.3866**
	RunTime (s)	0.059	0.141	33.49	35.55	687.78	0.731
Mediamill	AUC	0.7938	0.7793	0.7918	0.7563	0.7705	**0.7969**
	Macro-F1	0.0982	0.1266	0.1399	0.1269	0.1298	**0.1413**
	Micro-F1	0.5785	0.6422	0.6381	0.6273	0.6412	**0.6476**
	Accuracy	0.4264	0.4265	0.4326	0.4509	0.4465	**0.4691**
	RunTime (s)	0.278	10.09	4928.37	2534.60	8953.65	0.790
Corel5k	AUC	0.5534	0.5547	**0.5786**	0.5317	0.5573	0.5762
	Macro-F1	0.0383	0.0411	0.0273	0.0419	0.0422	**0.0497**
	Micro-F1	0.2241	0.2220	0.2230	0.2305	0.2322	**0.2700**
	Accuracy	0.1256	0.1162	0.1332	0.1447	0.1306	**0.1566**
	RunTime (s)	1.53	2.67	17021.46	1441.99	3957.35	15.36
SUN	AUC	0.7020	0.6950	0.6753	0.6760	0.6661	**0.7126**
	Macro-F1	0.2196	0.2630	0.1923	0.2507	**0.2852**	0.2687
	Micro-F1	0.4605	0.4936	0.4441	0.4670	0.4521	**0.5043**
	Accuracy	0.3009	0.3287	0.2877	0.3054	0.2954	**0.3388**
	RunTime (s)	0.2050	1.1104	571.69	1927.21	4016.15	0.7182

Table 2. Comparison of multi-label learning performance output by four algorithms on four real world multimedia datasets. (The best results are marked in dark and the second ones are underlined for each dataset.)

tuned from the candidate set $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}\}$. For generating the edge matrix $\mathbf{A}$ in SLRM, the number of nearest neighbors is set to 5, i.e., $c = 5$. The number of groups in MLLOC and the size of compressed label space in MIML are tuned with the same method as b in CPLST. The hyper-parameters in MIML are assigned according to the experimental setting in [29]. The parameters of these methods are tuned by conducting 10-fold cross-validation on the training set.

Based on the predicted label $\mathbf{C} \in \mathbb{R}^{k \times n_u}$ and the ground truth $\mathbf{G} \in \mathbb{R}^{k \times n_u}$ for n_u testing instances, the prediction performance are evaluated with the widely-used metrics in the field of multi-label classification, i.e., label-based Macro-F1, Micro-F1, instance-based Accuracy and AUC [33]. The former three measures require predefining a threshold to determine the number of labels for testing data. Here the number of labels for each testing instance is set according to its ground truth, i.e., $|\mathbf{C}_{\cdot,i}| = |\mathbf{G}_{\cdot,i}|$. In general, it is hard to set a proper threshold value in real applications. AUC (the area under the Receiver Operating Characteristic ROC curve) is used. Here the ROC curve is plot with respect to different threshold values.

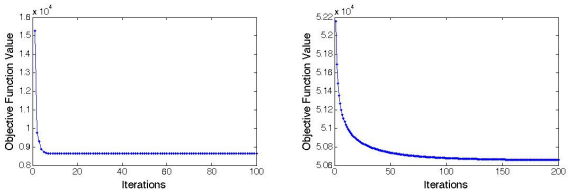
4.3. Results and Discussion

In the first experiment on *MSRC*, *Core5K*, *SUN* and *Mediamill* datasets, the instances in each class are evenly and randomly divided into 10 parts, one part for training and the rest for testing. For the class with less than 10 instances, we randomly select one instance for training and the rest for testing. We perform each method 10 runs on each data, and the average results are listed in Table 2. The best results of each evaluation measure are marked in bold, and the second best is underlined. According to the results, we can draw the following observations. (i) The proposed SLRM outperforms the supervised methods (CPLST, FAIE and MLLOC), which indicates that the unlabeled data are useful to learn the mapping function from a feature space and a label space. (ii) SLRM is superior to the semi-supervised method MC and active learning method MIML, which shows that SLRM provides a more reasonable strategy (the combination of a linear least square loss function with nuclear norm and a manifold regularizer) to obtain the best of both labeled and unlabeled data for effective exploiting label correlations and intrinsic geometric structure among data.

4.3.1 Convergency

In order to investigate the convergence of the algorithm to solve SLRM model, we plot the value of objective function (9) on two large datasets (*Corel5K* and *Mediamill*) in Figure 1. It can be seen that the objective function value decreases with respect to iterations, and the value approaches to be a fixed value after a few iterations (less than 10 iterations for *Corel5K* and less than 100 iterations for *Mediamill*).

The average running time of six methods on all datasets are listed in Table 2. By comparing with MC and MIML, the proposed SLRM method is pretty good in terms of computational complexity. MIML has to re-train the classifier after selecting the annotated points in each iteration, which is very time consuming. MC handles the joint matrix con-

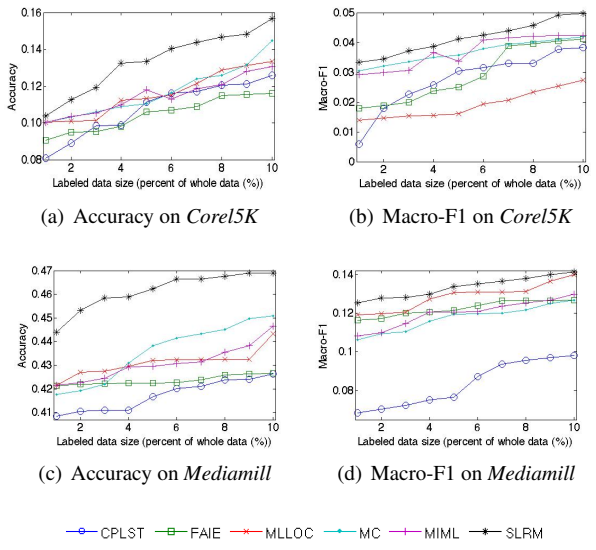


(a) *Corel5K* (b) *Mediamill*

Figure 1. Convergence of SLRM on *Corel5K* and *Mediamill*.

catenating the feature information $\mathbf{X}$ and label information $\mathbf{Y}$, thus it costs much more computational time. In the three supervised methods, both CPLST and FAIE are efficient because they only consider the labeled data. The performance of these two methods degrade significantly when there are only a few training data (see the experimental results in Figure 2). When the dataset is large (e.g., *Mediamill*), FAIE becomes time-consuming because it has to find the singular vectors of an $n_l \times n_l$ matrix. As MLLOC has to learn the local encoding for each class based on training data, it is slower than the other methods especially when the size of labels is large (e.g. *Corel5K*).

4.3.2 Effect of Training Data Size



(a) Accuracy on *Corel5K* (b) Macro-F1 on *Corel5K*

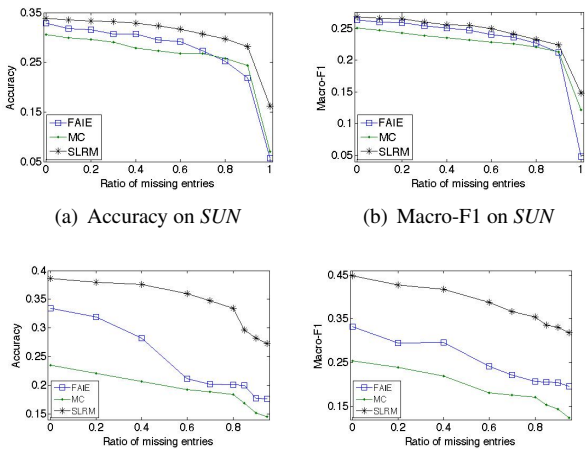
(c) Accuracy on *Mediamill* (d) Macro-F1 on *Mediamill*

Figure 2. Comparison of six methods under varying the labeled data sizes on *Corel5K* and *Mediamill*.

Here we studied the label prediction performance under varying the labeled data size. In the experiment, 1%-10% of data in each category are employed as training set. For a given percentage, a desired number of data are randomly sampled ten times, and the resulting average "instance-based Accuracy" and "label-based Macro-F1" on the unlabeled data are recorded. We plotted these results on two large datasets *Corel5K* and *Mediamill* in Figure 2. Obviously, the proposed method SLRM outperforms the other methods. It is interesting that SLRM performs well even when there are very few labeled data. This observation demonstrates that SLRM is more useful. In particular, it is very expensive to obtain more labeled data in real applications.

4.3.3 Handling Missing Label

Furthermore, we tested the performance of SLRM on handling missing labels. The experimental data are generated on datasets *SUN* and *MSRC* where the training and testing sets are the same with that in Table 2. For each training instance, we varied the ratio of missing labels (including positive and negative). In order to avoid the appearance of empty category and the instance with only negative labels, at least one instance is kept for each category and at least one positive label is kept for each instance. Then the label vector for training instance is set via $\mathbf{Y}_{j,i} = 0$ if the (j,i)th entry is missing, otherwise $\mathbf{Y}_{j,i} = 1$ if the ith instance belongs to the jth class, and $\mathbf{Y}_{j,i} = -1$ if the ith instance is not in the jth class. SLRM is compared with FAIE and MC, which are also capable of handling missing labels, and the results are given in Figure 3. SLRM clearly shows superior performance over other methods, especially on *MSRC*. We remark that handling *MSRC* is more difficult than handling *SUN* when partial labels are missing, because *MSRC* has less average labels in each instance than *SUN*.



(a) Accuracy on *SUN* (b) Macro-F1 on *SUN*

(c) Accuracy on *MSRC* (d) Macro-F1 on *MSRC*

Figure 3. Comparison results under varying the ratio of missing entries in label matrix (Y) on *SUN* and *MSRC*.

Meanwhile, we empirically validated our contribution that the mapping $\mathbf{U}$ has an ability to capture the label correlations. Since $\mathbf{U}$ indicates the low rank representation between features and labels, the correlations between labels can be measured by computing $\mathbf{U}^T\mathbf{U} \in \mathbb{R}^{k\times k}$. Ta-

Category	Related	Category	Related	Category	Related	Category	Related
aeroplane	road(0.157) sky(0.133)	building	body(0.231) car(0.217)	face	body(0.425) building(0.231)	sheep	grass(0.072) tree(0.066)
bicycle	tree (0.057) building(0.050)	car	building(0.152) road(0.143)	flower	face(0.119) grass(0.081)	sign	road(0.089) building(0.067)
bird	building(0.094) grass(0.074)	cat	road(0.037) grass(0.020)	grass	cow(0.168) sky(0.123)	sky	tree(0.256) road(0.242)
boat	water(0.166) tree(0.142)	chair	grass(0.042) building(0.039)	horse	grass(0.016) tree(0.015)	tree	road(0.271) sky(0.256)
body	face(0.425) building(0.217)	cow	grass(0.218) tree(0.106)	mountain	water(0.084) boat(0.056)	water	boat(0.168) tree(0.142)
book	face(0.133) body(0.133)	dog	road(0.069) body(0.058)	road	tree(0.271) sky(0.242)		

Table 3. Demonstration of label correlation identified by SLRM on MRSC data.

TRUE:	flower	TRUE:	aeroplane, grass, sky	TRUE:	dog, grass, tree, body	TRUE:	building, sky, road, tree
SLRM:	flower	SLRM:	aeroplane, grass, sky	SLRM:	dog, grass, face, body	SLRM:	building, car, road, tree
CPLST:	grass	CPLST:	building, sky, tree	CPLST:	cow, grass, tree, sky	CPLST:	building, grass, tree, sky
FAIE:	grass	FAIE:	building, road, sky	FAIE:	grass, body, face, road	FAIE:	building, sky, road, tree
MLLOC:	grass	MLLOC:	grass, road, sky	MLLOC:	grass, tree, road, sky	MLLOC:	grass, sky, road, tree
MC:	grass	MC:	building, grass, sky	MC:	building, grass, road, sky	MC:	building, sky, road, grass
MIML:	grass	MIML:	building, road, sky	MIML:	cow, grass, tree, sky	MIML:	building, sky, grass, tree

Figure 4. Image label prediction examples from MRSC data.

ble 3 gives the top two related categories for each category in MSRC. As expected, the returned categories are semantically related to the given category. Figure 4 lists four examples. The first two images are correctly labeled by SLRM while other methods can not. In the later two images, SLRM gave partial correct labels like other methods. As marked in red circle and red labels below the images, however, SLRM output the labels which are actually related to the image contents. Therefore, we can say that the proposed SLRM method is superior to state-of-the-art methods on multi-label classification.

5. Conclusions and Future Work

For tackling multi-label classification problems, in this paper, we have proposed a new model SLRM to identify an effective mapping function from a feature space to a label space. The proposed SLRM model can capture the label correlations by enforcing nuclear norm regularization on mapping function. SLRM also makes use of amounts of unlabeled data to smooth the mapping function by considering the intrinsic geometric structure among. In order to deal with large-scale data, an efficient algorithm based on alternating direction method of multipliers is developed to solve the proposed model. A series of experiments empirically demonstrated that SLRM was superior to state-of-the-art methods under varying the labeled data size and ratio of missing labels, and indicated that the proposed algorithm was efficient to predict label information for large-scale multi-label data.

In the future, it would be interesting to consider nonlinear loss function such as hinge loss and logistic loss instead of the current linear square loss function to measure the label approximation error of labeled data.

Acknowledgements

This work was supported by the NSFC Grant 61375062, the Ph.D. Programs Foundation of Ministry of Education of China Grant 20120009110006, PCSIRT Grant IRT201206, and HKRGC Grant GRF HKBU202013.

References

[1] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for attribute-based classification. In *Proc. of CVPR*, pages 612–620, 2013.

[2] A. Argyriou, T. Evgeniou, and M. Pontil. Convex multi-task feature learning. *Machine Learning*, 73:243–272, 2008.

[3] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning and Research*, 7:2399–2434, 2006.

[4] S. Bengio, J. Weston, and D. Grangier. Label embedding trees for large multi-class tasks. In *Proc. of NIPS*, pages 1–9, 2010.

[5] R. Cabral, F. Torre, J. Costeira, and A. Bernardino. Matrix completion for weakly supervised multi label image classification. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 37(1):121–135, 2015.

[6] D. Cai, X. He, J. Han, and T. Huang. Graph regularized non-negative matrix factorization for data representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011.

[7] J. Cai, E. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. Optimization*, 2(2):569–592, 2009.

[8] Y. Chen and H. Lin. Feature-aware label space dimension reduction for multi-label classification. In *Proc. of NIPS*, 2012.

[9] K. Dembczynski, W. Cheng, and E. Hullermeier. Bayes optimal multilabel classification via probabilistic classifier chains. In *Proc. of ICML*, pages 279–286, 2010.

[10] K. Dembczynski, W. Waegeman, W. Cheng, and E. Hullermeier. On label dependence and loss minimization in multi-label classification. *Machine Learning*, 88(1):5–45, 2012.

[11] G. Fung and O. Mangasarian. Multicategory proximal support vector machine classifiers. *Machine Learning*, 59(1):77–97, 2005.

[12] N. Ghamrawi and A. Mccallum. Collective multilabel classification. In *Proc. of CIKM*, pages 195–200, 2005.

[13] B. Hariharan, L. Zelnik-Manor, S. Vishwanathan, and M. Varma. Large scale max-margin multi-label classification with priors. In *Proc. of ICML*, pages 423–430, 2010.

[14] D. Hsu, S. Kakade, J. Langford, and T. Zhang. Multi-label prediction via compressed sensing. In *Proc. of NIPS*, 2009.

[15] S. Huang and Z. Zhou. Multi-label learning by exploiting label correlations locally. In *Proc. of AAAI*, 2012.

[16] X. Li and Y. Guo. Active learning with multi-label svm classification. In *Proc. of IJCAI*, 2013.

[17] Z. Lin, M. Chen, L. Wu, and Y. Ma. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. Technical report, UIUC Technical Report UILU-ENG-09-2215, 2009.

[18] Z. Lin, G. Ding, M. Hu, and J. Wang. Multi-label classification via feature-aware implicit label space encoding. In *Proc. of ICML*, 2014.

[19] Z. Lin, R. Liu, and Z. Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *Proc. of NIPS*, pages 612–620, 2011.

[20] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma. Robust recovery of subspace structures by low-rank representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013.

[21] M. Ng, F. Wang, and X. Yuan. Inexact alternating direction methods for image recovery. *SIAM J. on Scientific Computing*, 34(3):1643–1668, 2011.

[22] G. Patterson, C. Xu, H. Su, and J. Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108:59–81, 2014.

[23] M. Razaviyayn, M. Hong, and Z. Luo. A unified convergence analysis of block successive minimization methods for non-smooth optimization. *SIAM J. on Optimization*, 23(2):1126–1153, 2013.

[24] J. Rousu, C. Saunders, S. Szedmak, and J. Shawe-Taylor. Learning hierarchical multi-category text classifcation models. In *Proc. of ICML*, pages 774–751, 2005.

[25] N. Srebro, J. Rennie, and T. Jaakkola. Maximum-margin matrix factorization. In *Proc. of NIPS*, pages 1329–1336, 2004.

[26] L. Sun, S. Ji, and J. Ye. Hypergraph spectral learning for multi-label classification. In *Proc. of SIGKDD*, pages 668–676, 2008.

[27] F. Tai and H. Lin. Multi-label classification with principle label space transformation. In *Proc. of ICML*, 2010.

[28] G. Tsoumakas, I. Katakis, and I. Vlahavas. Random k-labelsets for multi-label classification. *IEEE Trans. on Knowledge and Data Engineering*, 23(7):1079–1089, 2011.

[29] D. Vasisht, A. Damianou, M. Varma, and A. Kapoor. Active learning for sparse bayesian multi-label classification. In *Proc. of SIGKDD*, 2014.

[30] H. Yu, P. Jain, P. Kar, and I. Dhillon. Large-scale multi-label learning with missing labels. In *Proc. of ICML*, pages 17–26, 2014.

[31] M. Yuan, A. Ekici, Z. Lu, and R. Monteiro. Dimension reduction and coefficient estimation in multivariate linear regression. *J. R. Statist. Soc*, 69(3):329–346, 2007.

[32] L. Zelnik and P. Perona. Self-tuning spectral clustering. In *Proc. of NIPS*, pages 1601–1608, 2004.

[33] M. Zhang and Z. Zhou. A review on multi-label learning algorithms. *IEEE Trans. on Knowledge and Data Engineering*, 26(8):1819–1837, 2014.

[34] Y. Zhang and J. Schneider. Multi-label output codes using canonical correlation analysis. In *Proc. of AISTATS*, pages 873–882, 2011.

[35] D. Zhou, O. Bousquet, T. Lai, J. Weston, and B. Scholkopf. Learning with local and global consistency. In *Proc. of NIPS*, 2003.

[36] H. Zou, J. Zhu, and T. Hastie. New multicategory boosting algorithms based on multicategory fisher consistent losses. *Ann Appl. Stat*, 2(4):1290–1306, 2008.