Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

Dictionary Learning with Mutually Reinforcing Group-Graph Structures

Hongteng Xu[1*], **Licheng Yu**[2*], **Dixin Luo**[3], **Hongyuan Zha**[4,5], **Yi Xu**[3]
[1]School of ECE, Georgia Institute of Technology, Atlanta, GA, USA
[2]Department of Computer Science, University of North Carolina at Chapel Hill, NC, USA
[3]SEIEE, Shanghai Jiao Tong University, Shanghai, China
[4]Software Engineering Institute, East China Normal University, Shanghai, China
[5]College of Computing, Georgia Institute of Technology, Atlanta, GA, USA

Abstract

In this paper, we propose a novel dictionary learning method in the semi-supervised setting by dynamically coupling graph and group structures. To this end, samples are represented by sparse codes inheriting their graph structure while the labeled samples within the same class are represented with group sparsity, sharing the same atoms of the dictionary. Instead of statically combining graph and group structures, we take advantage of them in a mutually reinforcing way — in the dictionary learning phase, we introduce the unlabeled samples into groups by an entropy-based method and then update the corresponding local graph, resulting in a more structured and discriminative dictionary. We analyze the relationship between the two structures and prove the convergence of our proposed method. Focusing on image classification task, we evaluate our approach on several datasets and obtain superior performance compared with the state-of-the-art methods, especially in the case of only a few labeled samples and limited dictionary size.

Introduction

Dictionary learning is the core of sparse representation models and helps to effectively reveal underlying structure in the data. Take image classification as an example. Learning a dictionary to allow sparse representation of images can capture high-level semantics of images. Generally, the discriminative power of sparse coding is highly correlated with the structure of dictionary, so several approaches have been pursued for dictionary learning. Among them, group sparsity (Bengio et al. 2009; Wang et al. 2011; Deng, Yin, and Zhang 2013) and Laplacian graph (Gao et al. 2010; Zheng et al. 2011; Long et al. 2013) are two popular regularization methods for learning the structure of the dictionary.

Group sparsity. Models exploiting group sparsity aim to encode samples with the same label using the same set of atoms of the dictionary. Each subset of dictionary atoms constructs a basis for the corresponding group. Compared with previous works in (Mairal et al. 2008; Wright et al. 2009; Jiang, Lin, and Davis 2011), which learn disjoint sub-dictionaries for representing different classes, the group sparse model has a huge advantage for large-scale classification problem — theoretically, the number of classes it can handle increases exponentially with the size of dictionary.

Laplacian graph. Laplacian graph captures the geometrical structure of the whole sample space by measuring the similarity among samples regardless of their labels. This method has been widely used in unsupervised learning, e.g., nonlinear dimension reduction, and semi-supervised learning, e.g., graph regularized sparse model (Gao et al. 2010; Zheng et al. 2011). Especially in the semi-supervised case, using graph-based label propagation algorithms (Goldberg et al. 2011), we can add new labels to samples.

These two structure regularization methods are complementary in nature: On one hand, training group sparse representation model relies on sufficient labeled samples, which can be obtained by graph-based label propagation. On the other hand, the group structure can introduce label information into graph, which prevents samples from being aggregated incorrectly. To our knowledge, there are few prior works leveraging both in a clean unified fashion.

Our main contribution is a method for dictionary learning over semi-supervised data that uses dynamic group and graph structures. We refer to these together as group-graph structures. The learning algorithm is partitioned into three phases: sparse coding, structure updating and dictionary updating. First, we take the group and graph regularization into account jointly for sparse coding. Second, we allow the group and graph structures to reinforce each other. An entropy-based label propagation strategy is proposed to incorporate several unlabeled samples into groups. These newly introduced samples are used to update their local graphs. Third, the dictionary is learned via the revised group-graph structures. We prove the convergence of the proposed method, and study the configurations of critical parameters. The image classification experiments on a variety of datasets show the superior performances of the proposed method compared to the state-of-art methods, especially in the case of a few labeled samples and limited dictionary size.

Proposed Model

The basic sparse representation model is shown as follows,

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2+\lambda\|\mathbf{A}\|_1, \tag{1}$$

*These two authors contribute equally.

where $\mathbf{X}$ is data matrix, whose columns are samples. $\mathbf{D} \in \mathbb{R}^{m\times K}$ is the dictionary we want to learn. $\mathbf{A}$ is the sparse code obtained during training process. Here $\|\cdot\|_F$ is the Frobenius norm of matrix. $\|\mathbf{A}\|_1$ is the absolute sum of the elements of $\mathbf{A}$.

Group Structure. Suppose that $\mathbf{X_L} \in \mathbb{R}^{m\times N_L}$ is a set of N_L labeled samples, which can be categorized into G classes $\mathbf{X_L} = [\mathbf{X}_1, \mathbf{X}_2, ..., \mathbf{X}_G]$. The group sparse model is shown as follows,

$$\min_{\mathbf{D},\mathbf{A_L}} \sum\nolimits_{g=1}^{G} \{\|\mathbf{X}_g - \mathbf{D}\mathbf{A}_g\|_F^2 + \lambda_1 \|\mathbf{A}_g\|_{1,2}\}, \quad (2)$$

where the coefficient matrix $\mathbf{A_L} = [\mathbf{A}_1, ..., \mathbf{A}_G] \in \mathbb{R}^{K\times N_L}$, and each $\mathbf{A}_g \in \mathbb{R}^{K\times N_g}$ ($\sum_{g=1}^{G} N_g = N_L$). The mixed l_1/l_2 norm $\|\mathbf{A}_g\|_{1,2} = \sum_{k=1}^{K} \|\mathbf{A}_g^k\|_2$ is the group sparsity regularization, where $\mathbf{A}_g^k$ is the k-th row of $\mathbf{A}_g$. It ensures samples within the same group are represented by the same basis from dictionary. For expression convenience, we further define group indicating matrices $\{\mathbf{S}_g \in \mathbb{R}^{N_L\times N_g}\}_{g=1}^{G}$ where each $\mathbf{S}_g$ is a binary matrix, indicating which columns of $\mathbf{A_L}$ belong to $\mathbf{A}_g$ for representing each $\mathbf{X}_g$, e.g., $\mathbf{A}_g = \mathbf{A_L}\mathbf{S}_g$.

Graph Structure. Given $\mathbf{X} = [\mathbf{X_L}, \mathbf{X_U}] \in \mathbb{R}^{m\times N}$ that contains both labeled samples and unlabeled ones, we construct a graph $\mathcal{G}$ whose vertices are $\mathbf{X}$. For any pair $\mathbf{x}_i$ and $\mathbf{x}_j$ in $\mathbf{X}$, if $\mathbf{x}_i$ is among p-nearest neighbors of $\mathbf{x}_j$ or vice versa, the weight of edge $w_{ij} = 1$, otherwise, $w_{ij} = 0$. All the weights formulate a weight matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$. The Laplacian graph matrix is $\mathbf{\Phi} = \text{diag}(d_1, ..., d_N) - \mathbf{W}$, where $d_i = \sum_{j=1}^{N} w_{ij}$. Then the graph-based sparse representation model can be described as follows,

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_1\|\mathbf{A}\|_1 + \lambda_2 \text{Tr}(\mathbf{A}\mathbf{\Phi}\mathbf{A}^T), \quad (3)$$

where the coefficient matrix $\mathbf{A} = [\mathbf{A_L}, \mathbf{A_U}]$. The learned dictionary ensures that the sparse codes preserve the local similarity of samples.

The Relationship between The Two Structures

The group regularizer $\sum_{g=1}^{G} \|\mathbf{A}_g\|_{1,2}$ in Eq. 2 ensures the sparsity of coefficient matrix and constrains the location of nonzero elements simultaneously. It actually can be interpreted as a graph regularizer with label-based metric. According to (Szlam, Gregor, and LeCun 2012; Tierney, Gao, and Guo 2014), we can rewrite $\sum_{g=1}^{G} \|\mathbf{A}_g\|_{1,2}$ as

$$\begin{aligned}&\sum\nolimits_{g=1}^{G} \{\|\mathbf{A}_g\|_1 + \gamma \sum\nolimits_{i,j\in\mathcal{C}_g} \|f(\mathbf{a}_i^L) - f(\mathbf{a}_j^L)\|_F^2\} \quad (4)\\ =&\|\mathbf{A_L}\|_1 + \gamma \text{Tr}(f(\mathbf{A_L})\mathbf{\Phi_L} f(\mathbf{A_L})^T).\end{aligned}$$

Here $\mathbf{a}_i^L$ denotes the column of $\mathbf{A_L}$. $\mathcal{C}_g$ indicates the indices of samples belonging to the gth group. Function $f(\cdot)$ is applied to the elements of $\mathbf{A_L}$: $f(a) = 1$ if $a \neq 0$, otherwise $f(a) = 0$. It projects coefficient matrix into the space of its structure. $\mathbf{\Phi_L} \in \mathbb{R}^{N_L\times N_L}$ is a block diagonal matrix, which is a label-based Laplacian graph matrix:

$$\mathbf{\Phi_L} = diag(\mathbf{\Phi}_1, ..., \mathbf{\Phi}_G),\ \mathbf{\Phi}_g(i,j) = \begin{cases} N_g, & i = j, \\ -1, & i \neq j. \end{cases} \quad (5)$$

$\mathbf{\Phi_L}$ corresponds to the union of G complete graphs - for each labeled sample, all the rest having the same label are its neighbors. From this view, the group sparsity is equivalent to regularize the structure of coefficient matrix with a label-based graph. Note that there are two differences between group sparsity with the Maximum Mean Discrepancy (MMD) (Gretton et al. 2006) constraint: 1) group sparsity is purely based on label information, it does not add empirical assumption to the relationship between labeled and unlabeled samples; 2) the label-based graph regularizes the structure of coefficient matrix $f(\mathbf{A_L})$, rather than $\mathbf{A_L}$ itself.

Learning with Mutually Reinforcing Group-Graph Structures

As we mentioned before, the group structure fails to regularize unlabeled ones, while the graph structure might incorrectly connect samples from different classes. A naive way to tackle this thorny issue is combining these two structures directly as follows,

$$\min_{\mathbf{A},\mathbf{D}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda \sum_{g=1}^{G} \|\mathbf{A_L}\mathbf{S}_g\|_{1,2} + \mu \text{Tr}(\mathbf{A}\mathbf{\Phi}\mathbf{A}^T). \quad (6)$$

Unfortunately, such a simple combination is often useless. Group sparsity is still limited on labeled samples, which has little influence on the graph structure of unlabeled samples. The graph structure provides little information for grouping as well. In summary, we need to establish connections between these two structures. To address this problem, we propose the following sparse representation model with mutually reinforcing group-graph structures.

$$\begin{aligned}\min_{\mathbf{A},\mathbf{D},\mathbf{\Phi},\{\mathbf{S}_g\}_{g=1}^{G}} &\|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda_1 \sum\nolimits_{i=1}^{N-N_L} \|\mathbf{a}_i^U\|_1 \quad (7)\\ &+\lambda_2 \sum\nolimits_{g=1}^{G} \|\mathbf{A_L}\mathbf{S}_g\|_{1,2} + \mu \text{Tr}(\mathbf{A}\mathbf{\Phi}\mathbf{A}^T),\end{aligned}$$

where $\mathbf{a}_i^U \in \mathbf{A_U}$ is the i-th unlabeled sample. The first term in Eq. (7) measures the reconstruction error, the second ensures sparse representation for unlabeled samples, and the group sparsity in the third term pursues the group sparse representation for labeled samples. The last term is the graph regularization for all sparse codes. The most significant difference between Eq. (6) and Eq. (7) is besides $\mathbf{D}$ and $\mathbf{A}$, we also update $\{\mathbf{S}_g\}_{g=1}^{G}$ and $\mathbf{\Phi}$ in Eq. (7) to optimize the group and graph structures. In other words, the group and graph structures are updated during dictionary learning.

The Learning Algorithm

The optimization of Eq. (7) is challenging, which involves non-convex optimization ($\mathbf{D}$ and $\mathbf{A}$) and dynamic programming ($\{\mathbf{S}_g\}_{g=1}^{G}$ and $\mathbf{\Phi}$). In this paper, we propose an effective algorithm, dividing the problem into the following three subproblems and solving them iteratively.

- **Sparse coding:** Fix $\mathbf{D}$, $\{\mathbf{S}_g\}_{g=1}^{G}$ and $\mathbf{\Phi}$, optimize $\mathbf{A}$.
- **Structure updating:** According to $\mathbf{A}$, update $\{\mathbf{S}_g\}_{g=1}^{G}$ and $\mathbf{\Phi}$ alternatively by entropy based label propagation.
- **Dictionary updating:** Given $\mathbf{A}$, optimize $\mathbf{D}$.

Joint Group-Graph Sparse Coding

In sparse coding phase, we aim to obtain $\mathbf{A}$ with fixed $\mathbf{D}$, $\{\mathbf{S}_g\}_{g=1}^G$ and $\mathbf{\Phi}$. Solving Eq. (7) directly is time-consuming, so similar to (Tropp, Gilbert, and Strauss 2006; Szlam, Gregor, and LeCun 2012) we replace the mixed $l_{1,2}$ norm in Eq. (7) with $l_{0,\infty}$ norm, and impose sparsity constraints explicitly. The optimization problem is rewritten as

$$\begin{aligned} \min_{\mathbf{A}} \ & \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2+\mu\mathrm{Tr}(\mathbf{A}\mathbf{\Phi}\mathbf{A}^T). \\ \text{s.t.} \ & \|\mathbf{A}_g\|_{0,\infty}\le C, \quad g=1,...,G. \\ & \|\mathbf{a}_i^U\|_0\le C, \quad \mathbf{a}_i^U\in\mathbf{A}_\mathbf{U}. \end{aligned} \tag{8}$$

where $\|\mathbf{A}_g\|_{0,\infty}$ counts the number of rows having nonzero elements in $\mathbf{A}_g$, and $\|\mathbf{a}_i^U\|_0$ counts the number of nonzero elements in $\mathbf{a}_i^U$, both of which are bounded by C. We solve this problem by introducing an auxiliary variable $\mathbf{Z}$ to approximate $\mathbf{A}$. Then, Eq. (8) can be expressed as follows,

$$\begin{aligned} \min_{\mathbf{A},\mathbf{Z}} \ & \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2+\mu\mathrm{Tr}(\mathbf{Z}\mathbf{\Phi}\mathbf{Z}^T)+\beta\|\mathbf{A}-\mathbf{Z}\|_F^2, \\ \text{s.t.} \ & \|\mathbf{A}_g\|_{0,\infty}\le C, \quad g=1,...,G, \\ & \|\mathbf{a}_i^U\|_0\le C, \quad \mathbf{a}_i^U\in\mathbf{A}_\mathbf{U}, \end{aligned} \tag{9}$$

which can be solved by splitting into two parts and optimizing $\mathbf{A}, \mathbf{Z}$ alternatively.

[A-part]: We fix $\mathbf{Z}$ to optimize $\mathbf{A}$ by

$$\begin{aligned} \min_{\mathbf{A}} \ & \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2+\beta\|\mathbf{Z}-\mathbf{A}\|_F^2, \\ \text{s.t.} \ & \|\mathbf{A}_g\|_{0,\infty}\le C, \quad g=1,...,G, \\ & \|\mathbf{a}_i^U\|_0\le C, \quad \mathbf{a}_i^U\in\mathbf{A}_\mathbf{U} \quad , \end{aligned} \tag{10}$$

whose objective function can be rewritten as $\min_{\mathbf{A}}\|\binom{\mathbf{X}}{\sqrt{\beta}\mathbf{Z}}-\binom{\mathbf{D}}{\sqrt{\beta}\mathbf{I}}\mathbf{A}\|_F^2$. We view each $\mathbf{a}_i^U$ as a "group" with only one member so that Eq. (10) becomes a group sparse coding problem, which can be efficiently solved using simultaneous orthogonal matching pursuit (S-OMP) (Tropp, Gilbert, and Strauss 2006).

[Z-part]: We approximate $\mathbf{Z}$ with the fixed $\mathbf{A}$,

$$\min_{\mathbf{Z}}\beta\|\mathbf{Z}-\mathbf{A}\|_F^2+\mu\mathrm{Tr}(\mathbf{Z}\mathbf{\Phi}\mathbf{Z}^T). \tag{11}$$

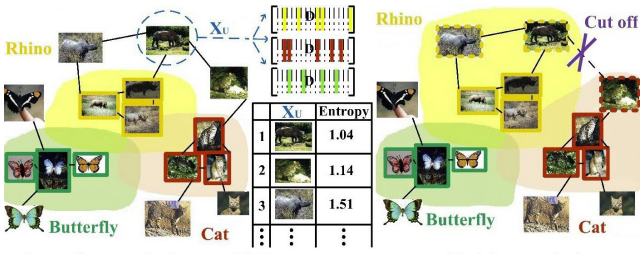
Here, $\mathbf{Z}$ can be efficiently computed using gradient descent, whose update equation is

$$\mathbf{Z}_{t+1}=\mathbf{Z}_t+\nu[\mathbf{Z}_t(\mu\mathbf{\Phi}+\beta\mathbf{I})-\beta\mathbf{A}], \tag{12}$$

where $\mathbf{Z}_t$ is the t-th estimation, and ν is the step size of the gradient descent. With the progressively increased weight β in each iteration, we can finally reach the optimal $\mathbf{A}$.

Group-Graph Structure Updates

The sparse codes computed above accomplish two important works: 1) the bases $\{\mathbf{D}_g\}_{g=1}^G$, which are used for representing groups $\{\mathbf{X}_g\}_{g=1}^G$, have been adaptively chosen according to non-zero positions of $\{\mathbf{A}_g\}_{g=1}^G$; 2) the local similarity of samples $\mathbf{X}$ has been inherited by sparse codes. In contrast to other dictionary learning methods that update dictionary directly after sparse coding, we propose an updating method for improving both group structure $\{\mathbf{S}_g\}_{g=1}^G$ and graph structure $\mathbf{\Phi}$ — propagating labels to several unlabeled samples and updating graph accordingly.



(a) Before Updating (b) Compute entropy (c) After Updating

Figure 1: (a) Before updating, the labeled samples $\mathbf{X_L}$ are outlined with different colors, and the unlabeled samples are linked to them by graph. (b) The unlabeled samples $\mathbf{X_U}$ are projected onto the bases for computing the reconstruction errors, and are sorted according to entropy values. (c) The unlabeled samples with high confidence are introduced into groups, and their nearby graphs are updated.

Group Update. For each $\mathbf{x}_i^U\in\mathbf{X_U}$, we have calculated its coefficient vector $\mathbf{a}_i^U$ in the sparse coding phase. For identifying its label, we can follow the reconstruction-based criteria in (Wright et al. 2009; Yang, Zhang, and Feng 2011) by extracting the coefficients $\mathbf{a}_{i,g}^U$ from $\mathbf{a}_i^U$ that are associated with basis $\mathbf{D}_g$[1] and calculate its reconstruction error,

$$\mathrm{Err}_{i,g}^U=\|\mathbf{x}_i^U-\mathbf{D}_g\mathbf{a}_{i,g}^U\|_2^2. \tag{13}$$

After computing the errors on each basis, it is natural to identify its label as $\mathbf{Id}(\mathbf{x}_i^U)=\min_g\mathrm{Err}_{i,g}^U$. However, this strategy risks propagating labels incorrectly for the samples around decision boundary of two groups, whose reconstruction errors might be comparable.

Inspired by (Zhang, Jiang, and Davis 2013), we propose an entropy-based label propagation method for reducing the risk. Let $\mathrm{P}_{i,g}^U$ be the probability of $\mathbf{x}_i^U$ being in group g. In our work, we compute it as follows,

$$\mathrm{P}_{i,g}^U=\frac{(\mathrm{Err}_{i,g}^U+\epsilon)^{-1}}{\sum_{c=1}^G(\mathrm{Err}_{i,c}^U+\epsilon)^{-1}}, \tag{14}$$

where the positive parameter ϵ helps avoiding zero denominator. Then the uncertainty of label identification for $\mathbf{x}_i^U$ can be quantified using the entropy of $\{\mathrm{P}_{i,g}^U\}_{g=1}^G$,

$$\mathrm{E}_i^U=-\sum_{g=1}^G\mathrm{P}_{i,g}^U\log\mathrm{P}_{i,g}^U. \tag{15}$$

The lower entropy indicates that we can label $\mathbf{x}_i^U$ correctly with higher certainty. So, we sort the unlabeled samples in ascending entropy values, and incorporate top $\alpha\%$ into their predicted groups. Accordingly, the group indicating matrices $\{\mathbf{S}_g\}_{g=1}^G$ are updated.

Graph Update. Given the new group structure, we then update graph structure accordingly. The newly labeled sample should enlarge the inter-group distance and shrink the

[1] $\mathbf{a}_{i,g}^U$ is a vector whose elements are those in $\mathbf{a}_i^U$ corresponding to columns of $\mathbf{D}_g$.

intra-group distance simultaneously. Therefore, we cut off its connections to other groups and preserve its neighbors from the same group. As a result, we obtain a revised Laplacian graph matrix $\mathbf{\Phi}$, which will be used to regularize sparse codes in the next iteration and assist to update group structure implicitly. Compared with the Maximum Mean Discrepancy (MMD) regularization in (Long et al. 2013), our strategy is more flexible because the graph structure is updated during learning process. Even if the initial labeled samples are insufficient, after adding new labels in the following iterations, the influence of label information on graph structure will become more and more significant.

Fig. 1 gives an example for illustrating the updates. Using our approach, we learn a dictionary from Caltech101 dataset (Fei-Fei, Fergus, and Perona 2007). Fig. 1 shows the group-graph structures of images belonging to 3 classes. We can find that after updating structures, some unlabeled samples are labeled into groups and the inter-group connection is deleted. The label information is enhanced and the mistakes in the graph are corrected. With the help of the updates above, the graph and group structures reinforce mutually.

Dictionary Update

Given coefficient matrix $\mathbf{A}$ and samples $\mathbf{X}$, we can update dictionary $\mathbf{D}$ by solving the following optimization problem,

$$\begin{aligned} &\min_{\mathbf{D}} \ \|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2, \\ &\text{s.t. } \|\mathbf{d}_k\|_2 \le 1, \quad k=1,...,K. \end{aligned} \tag{16}$$

This has been well studied by previous works (Lee et al. 2006; Zheng et al. 2011), thus we omit the technical optimization details here. It should be mentioned that the remaining unlabeled samples are still used for dictionary learning, which serve as replenishing the structures of dictionary (Raina et al. 2007).

Convergence Analysis of the Algorithm

The three phases above are performed till the convergence of dictionary learning, so the feasibility of our algorithm depends on its convergence. As we mentioned before, the original problem Eq. (7) is composed of sparse coding, structure updating and dictionary updating. Because we decompose sparse coding problem Eq. (9) into two convex optimization problems Eq. (10) and Eq. (11), the convergence of sparse coding phase is guaranteed. Similarly, the dictionary updating is achieved by solving convex optimization problem Eq. (16). So, for demonstrating the convergence of our learning algorithm, we just need to prove that the structure updating decreases objective value monotonically.

Proposition 1 *Given* $\mathbf{A}$ *and* $\mathbf{D}$*, the updates of group structure* $\{\mathbf{S}_g\}_{g=1}^G$ *and graph structure* $\mathbf{\Phi}$ *reduce objective value in Eq. (7) monotonically.*

Proof Without loss of generality, we consider the case where one unlabeled sample $\mathbf{x}_i^U$ is labeled. We add it into the g-th group and cut its edges to the samples in other groups. The sparse code of $\mathbf{x}_i^U$ is $\mathbf{a}_i^U \in \mathbb{R}^K$, whose element is a_k^U. The sparse codes corresponding to the original g-th group is $\mathbf{A}_g = [a_{kn}] \in \mathbb{R}^{K\times N_g}$.

Group update. In Eq. (7), the group related term is changed from $\lambda_1\|\mathbf{a}_i^U\|_1 + \lambda_2\|\mathbf{A}_g\|_{1,2}$ to $\lambda_2\|[\mathbf{A}_g, \mathbf{a}_i^U]\|_{1,2}$. Here we assume $\lambda_1 = \lambda_2 = \lambda$[2]. Then, according to Jensen's inequality, we have

$$\begin{aligned} &\|[\mathbf{A}_g, \mathbf{a}_i^U]\|_{1,2} - (\|\mathbf{a}_i^U\|_1 + \|\mathbf{A}_g\|_{1,2}) \\ =&\sum_{k=1}^K \sqrt{(a_k^U)^2 + \sum_{n=1}^{N_g} a_{kn}^2} - \left(\sum_{k=1}^K |a_k^U| + \sum_{k=1}^K \sqrt{\sum_{n=1}^{N_g} a_{kn}^2}\right) \\ =&\sum_{k=1}^K \left(\sqrt{(a_k^U)^2 + \sum_{n=1}^{N_g} a_{kn}^2} - \left(\sqrt{(a_k^U)^2} + \sqrt{\sum_{n=1}^{N_g} a_{kn}^2}\right)\right) \\ \le&0. \end{aligned}$$

Graph update. The Laplacian graph matrix $\mathbf{\Phi} = \text{diag}(d_1, ..., d_N) - \mathbf{W}$, where $\mathbf{W} = [w_{ij}]$ is the 0-1 weight matrix defined before and $d_i = \sum_{j=1}^N w_{ij}$. Suppose that originally $\mathbf{x}_i^U$ is connected with a sample $\mathbf{x}_j$ not in the g-th group ($w_{ij} = 1$). After adding $\mathbf{x}_i^U$ into the g-th group, w_{ij} is set to be 0, so the new Laplacian graph can be written as $\mathbf{\Phi}_{\text{new}} = \mathbf{\Phi} + \Delta\mathbf{\Phi}$. Here $\Delta\mathbf{\Phi} = [\delta_{rc}]$ has only four nonzero elements: $\delta_{ii} = \delta_{jj} = -1$, $\delta_{ij} = \delta_{ij} = 1$. Obviously, $\Delta\mathbf{\Phi}$ is negative-semidefinte, so

$$\text{Tr}(\mathbf{A}\mathbf{\Phi}_{\text{new}}\mathbf{A}^T) - \text{Tr}(\mathbf{A}\mathbf{\Phi}\mathbf{A}^T) = \text{Tr}(\mathbf{A}\Delta\mathbf{\Phi}\mathbf{A}^T) \le 0.$$

In summary, after updating the group and graph structures, the objective value decreases monotonically. □

Classification

After computing $\mathbf{A_L}$ for both given labeled samples and propagated ones, we train a linear SVM classifier by following the method in (Zheng et al. 2011). When a testing sample $\mathbf{x_t} \in \mathbb{R}^m$ comes, we first search its p-nearest neighbors, denoted as $\widetilde{\mathbf{X}} \in \mathbb{R}^{m\times p}$, from samples $\mathbf{X}$. The corresponding coefficients of $\widetilde{\mathbf{X}}$ are found in $\mathbf{A}$, which is denoted as $\widetilde{\mathbf{A}}$. Then, the sparse codes of $\mathbf{x_t}$ is computed by solving the follows,

$$\begin{aligned} &\min_{\mathbf{a_t}} \ \|\mathbf{x_t} - \mathbf{D}\mathbf{a_t}\|_2^2 + \mu\|(\mathbf{a_t}\mathbf{1}^T - \widetilde{\mathbf{A}})\|_F^2, \\ &\text{s.t. } \|\mathbf{a_t}\|_0 \le C. \end{aligned} \tag{17}$$

where $\mathbf{1}$ is a vector whose elements are all 1's. This problem can be solved effectively by many sparse coding algorithms. Taking $\mathbf{a_t}$ as the input of SVM, we obtain its category. This method ensures that the feature of sample is robust to the small change of sample.

Related Work

Previous dictionary learning methods, like KSVD (Aharon, Elad, and Bruckstein 2006), aim to reconstruct data with high accuracy. Focusing on classification problem, we require the dictionary to be not only representative but also discriminative. Mairal *et al* first proposed a discriminative

[2]In following experiments, we set $\lambda_1 = \lambda_2$ indeed.

Table 1: Classification results for various numbers of labeled samples per class.

	Extended YaleB				UIUC-Sports				Scene15				Caltech101			
#Label	5	10	20	32	10	20	45	70	10	30	50	100	5	15	20	30
ScSPM	-	-	-	-	67.8	73.2	79.6	82.7	65.3	72.1	73.8	80.3	-	67.0	-	73.2
LLC	67.7	80.1	88.5	94.7	-	-	-	-	64.1	72.4	73.0	80.6	51.2	65.4	67.7	73.4
DKSVD	-	76.1	92.0	94.0	-	73.6	77.9	81.7	-	68.4	71.9	79.1	49.6	65.1	68.6	73.0
LCKSVD	-	74.5	92.4	94.2	-	73.4	77.6	82.9	-	70.3	73.1	79.7	54.0	67.7	70.5	73.6
GroupSC	64.7	81.2	92.4	94.2	-	-	-	-	-	-	-	-	52.0	65.8	67.9	72.7
SelfSC	72.3	83.7	90.3	93.6	68.9	73.5	77.9	81.0	59.7	69.5	70.8	76.8	53.5	64.9	66.3	68.8
TSC	73.2	84.1	91.6	94.5	70.1	74.0	78.9	82.9	63.4	71.9	72.7	80.0	55.2	66.9	68.2	70.6
Ours	**80.5**	**89.7**	**93.5**	**96.1**	**74.3**	**76.7**	**81.1**	**83.3**	**67.8**	**73.3**	**74.1**	**80.8**	**62.9**	**70.3**	**71.1**	**74.2**

KSVD in (Mairal et al. 2008), which introduced a label-related penalty function in the framework of KSVD. Following this way, many variants of KSVD appear, such as DKSVD (Zhang and Li 2010) and LCKSVD (Jiang, Lin, and Davis 2011). More recently, several methods combine sparse coding with transfer learning (Huang et al. 2013; Al-Shedivat et al. 2014), which ensure the learned dictionary to be more suitable for testing data. For learning more compact dictionary, group sparse representation is proposed in (Bengio et al. 2009; Gao et al. 2010; Chi et al. 2013).

Sparse-based classifier can also be learned in a semi-supervised way (Zhang et al. 2011). Self-taught learning methods (Raina et al. 2007; Lee et al. 2009) teach dictionary to learn abundant structures from unlabeled samples, and produce sparse codes for labeled samples. LaplacianSC (Gao et al. 2010), GraphSC (Zheng et al. 2011) and TSC (Lim, Torralba, and Salakhutdinov 2011) exploited Laplacian graph matrix to characterize sample similarity for sparse codes, achieving promising classification results. The basic idea is capturing the geometrical information (graph structure) of samples for regularizing model. These semi-supervised learning methods rely on using sparse codes of labeled samples to train classifier, e.g., SVM, Logistic regression, so they are still unable to cope with extremely few labels. What is worse, the graph structures in (Gao et al. 2010; Zheng et al. 2011; Lim, Torralba, and Salakhutdinov 2011) are constructed without label information, which might provide sparse codes with wrong local constraints. In (Long et al. 2013), the label-based Maximum Mean Discrepancy (MMD) (Gretton et al. 2006) constraint is combined with Laplacian graph. However, such a combination is statical. Without label propagation (Goldberg et al. 2011), the influence of label is limited. Recently, an online semi-supervised dictionary learning method is proposed in (Zhang, Jiang, and Davis 2013). It designs a label propagation method in the learning phase, but it does not take group structure into consideration.

Experiments

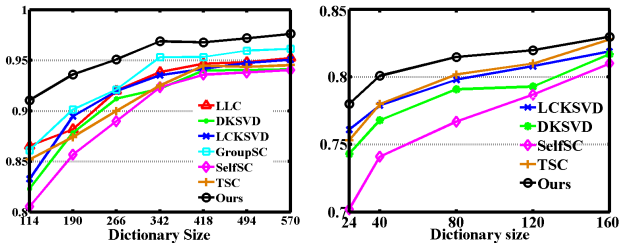
Experiments on Various Datasets

We compare the proposed method to prior sparsity-based methods, including **ScSPM** (Yang et al. 2009), **LLC** (Wang et al. 2010), **DKSVD** (Zhang and Li 2010), **LCKSVD** (Jiang, Lin, and Davis 2011), **GroupSC** (Bengio et al. 2009), **SelfSC**(Raina et al. 2007) and **TSC**(Long et al. 2013). Note that we artificially partition the data into two sets, and do not use the labels for one set in order to simulate the semi-supervised setting, i.e., SelfSC, TSC and ours. Following the configuration in (Jiang, Lin, and Davis 2011), we evaluate our method on four datasets: **1) Extended YaleB** (Georghiades, Kriegman, and Belhurneur 1998) contains 2414 frontal face images of 38 persons. We randomly select 5, 10, 20 and 32 samples per category as labeled samples, and another 32 samples as testing samples. The rest samples are used as unlabeled ones. The dictionary size is set to be $K = 380$ for all methods. **2) UIUC-sports** (Li and Fei-Fei 2007) consists of 8 sport event categories with 137 to 250 images in each. We randomly select 10, 20, 45 and 70 images for labeling, and another 60 images for testing, while the left ones are used as unlabeled samples. The dictionary size is set to be $K = 160$ for all methods. **3) Scene15** (Lazebnik, Schmid, and Ponce 2006) contains 15 categories and 4,485 images in all, 200 to 400 images per category. We randomly select 10, 30, 50 and 100 images per class for labeling and another 100 images for testing. The remaining images are used as unlabeled ones. The dictionary size is set to be $K = 450$ for all methods. **4) Caltech101** (Fei-Fei, Fergus, and Perona 2007) contains 9144 images from 102 classes, i.e., 101 object classes and a 'background' class. Like (Lazebnik, Schmid, and Ponce 2006), we randomly pick up 5, 15, 20 and 30 labeled samples per category and test on up to 50 images per class. The remaining unlabeled samples are used for semi-supervised learning. According to the number of labeled samples, we set dictionary size K to be 500, 800, 1000 and 1500 respectively for all methods.

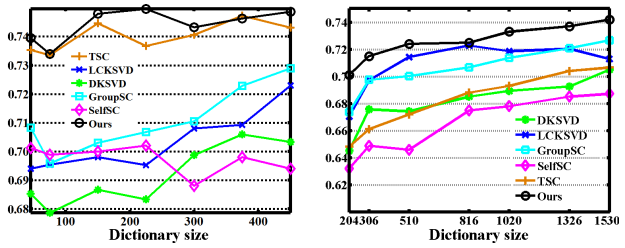
Other parameters are given as follows: Like (Zheng et al. 2011), the number of neighbors for each sample is set to be $p = 2$ in the graph construction; the percentage α for label propagation is set to be 10; the sparsity C are set according to the datasets — $C = 20$ for Extended YaleB, Scene15 and Caltech101, and $C = 25$ for UIUC-Sports; the graph weight μ is set to be 0.2 for Extended YaleB and Caltech101, and 0.5 for UIUC-Sports and Scene15. The rationality of these configurations will be analyzed in the next subsection.

The classification results[3] listed in Table 1 demonstrate that our method can achieve much higher classification accuracy than others, especially when merely few labeled samples are provided. Furthermore, we plot the performance

[3]All the classification accuracies reported in this paper are the averaged results of 5 repeated experiments.



(a) Extended YaleB (b) UIUC-sports



(c) Scene15 (d) Caltech101

Figure 2: Classification accuracy with different dictionary sizes. The numbers of labeled samples used in the test are 32 labeled samples/class for Extended YaleB, 70 labeled samples/class for UIUC-sports and Scene15, and 30 labeled samples/class for Caltech101.

curves of the methods w.r.t different dictionary sizes in Fig. 2. The experimental results show that even when the dictionary sizes are very limited, i.e., 114 for Extended YaleB, 40 for UIUC-sports, 45 for Scene15 and 204 for Caltech 101, the classification accuracies of our method are still above 90%, 80%, 74% and 70% respectively, which are much higher than those obtained by its competitors. These results prove that our method is superior to its competitors in the case of limited dictionary size.

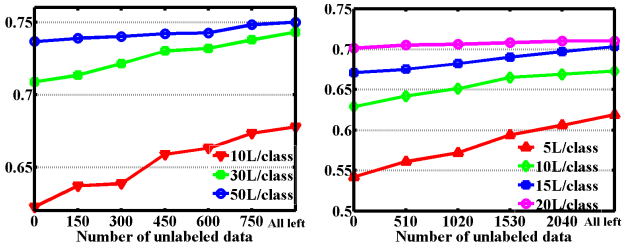
Influences of Other Factors

Besides the number of labeled samples and the dictionary size, there are several other important factors in our method. Firstly, we analyze the effect of the mutually reinforcing group-graph structure on classification results on Caltech101 in Table 2, where the number of unlabeled data being propagated and the propagation error rate are also listed. Compared with the method using static group-graph structure, our method achieves improvements on classification accuracy. The more samples we label rightly, the more improvements on classification accuracy we obtain.

Table 2: Effect of the dynamic group-graph structure.

#Label	5	10	15	20	25
Static structure(%)	60.5	66.3	69.7	70.7	72.8
Dynamic structure(%)	**62.9**	**67.3**	**70.3**	**71.1**	**72.9**
# sample propagated	368	276	188	102	42
Propagation error rate(%)	16.6	13.0	8.0	6.9	4.8

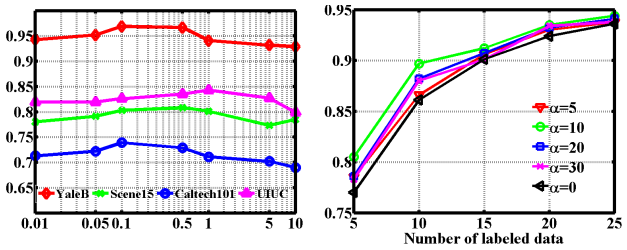
Secondly, we analyze the performance in terms of different number of unlabeled samples in our algorithm. The accuracy curves are plotted in Fig. 3, which show the learned dictionary can be more effective in discriminating between categories with more unlabeled samples provided.



(a) Scene15 (b) Caltech101

Figure 3: Influences of unlabeled samples on classification accuracy.

We also run our method with varying weight of graph regularization μ. Theoretically, large μ tends to enhance the similarity among the sparse codes of local samples. We plot the classification accuracies w.r.t different values of μ in Fig. 4(a) using full datasets. It is observed that our approach can be robust with its range from 0.1 to 0.5. We then investigate the effect of α on our algorithm using Extended YaleB dataset. Fig. 4(b) shows the performance of our algorithm w.r.t the percentage ($\alpha\%$) of unlabeled samples used for propagation during each iteration. It can be observed that small α might produce little effect on our method, as the number of newly labeled samples is too small to enhance the group structure. On the contrary, if we use large α, some unlabeled samples with low certainty will be introduced with high risks, in which case, the incorrectly labeled samples might produce negative effect on structure update.



(a) μ (b) α

Figure 4: Influences of parameter configurations on classification accuracy.

Conclusion

In this paper, we propose a dictionary learning algorithm with mutually reinforcing group-graph structure and demonstrate its convergence. During dictionary learning, the group and graph structures update via label propagation and graph modification for learning a more discriminative and robust dictionary. Based on theoretical analysis and experiments on various datasets, we prove the superiority of our method. Currently, some errors appear during label propagation, which degrade the benefits of dynamically updating

the structure (see Table 2). Future work will address reducing errors during propagation.

Acknowledgement: This work is supported in part by NSF grant DMS-1317424, and NSFC-61129001/F010403, 61025005.

References

Aharon, M.; Elad, M.; and Bruckstein, A. 2006. The k-svd: An algorithm for designing of overcomplete dictionaries for sparse representations. *Signal Processing, IEEE Transactions on* 54(11):4311–4322.

Al-Shedivat, M.; Wang, J. J.-Y.; Alzahrani, M.; Huang, J. Z.; and Gao, X. 2014. Supervised transfer sparse coding. In *AAAI*.

Bengio, S.; Pereira, F.; Singer, Y.; and Strelow, D. 2009. Group sparse coding. In *NIPS*, 82–89.

Chi, Y.-T.; Ali, M.; Rajwade, A.; and Ho, J. 2013. Block and group regularized sparse modeling for dictionary learning. In *CVPR*, 377–382. IEEE.

Deng, W.; Yin, W.; and Zhang, Y. 2013. Group sparse optimization by alternating direction method. In *SPIE Optical Engineering+ Applications*, 88580R–88580R. International Society for Optics and Photonics.

Fei-Fei, L.; Fergus, R.; and Perona, P. 2007. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding* 106(1):59–70.

Gao, S.; Tsang, I. W.; Chia, L.-T.; and Zhao, P. 2010. Local features are not lonely–laplacian sparse coding for image classification. In *CVPR*, 3555–3561. IEEE.

Georghiades, A. S.; Kriegman, D. J.; and Belhurneur, P. 1998. Illumination cones for recognition under variable lighting: Faces. In *CVPR*, 52–58. IEEE.

Goldberg, A. B.; Zhu, X.; Furger, A.; and Xu, J.-M. 2011. Oasis: Online active semi-supervised learning. In *AAAI*.

Gretton, A.; Borgwardt, K. M.; Rasch, M.; Schölkopf, B.; and Smola, A. J. 2006. A kernel method for the two-sample-problem. In *NIPS*, 513–520.

Huang, J.; Nie, F.; Huang, H.; and Ding, C. H. 2013. Supervised and projected sparse coding for image classification. In *AAAI*.

Jiang, Z.; Lin, Z.; and Davis, L. S. 2011. Learning a discriminative dictionary for sparse coding via label consistent k-svd. In *CVPR*, 1697–1704. IEEE.

Lazebnik, S.; Schmid, C.; and Ponce, J. 2006. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, volume 2, 2169–2178. IEEE.

Lee, H.; Battle, A.; Raina, R.; and Ng, A. Y. 2006. Efficient sparse coding algorithms. In *NIPS*, 801–808.

Lee, H.; Raina, R.; Teichman, A.; and Ng, A. Y. 2009. Exponential family sparse coding with application to self-taught learning. In *IJCAI*, volume 9, 1113–1119.

Li, L.-J., and Fei-Fei, L. 2007. What, where and who? classifying events by scene and object recognition. In *ICCV*. IEEE.

Lim, J. J.; Torralba, A.; and Salakhutdinov, R. 2011. Transfer learning by borrowing examples for multiclass object detection. In *NIPS*, 118–126.

Long, M.; Ding, G.; Wang, J.; Sun, J.; Guo, Y.; and Yu, P. S. 2013. Transfer sparse coding for robust image representation. In *CVPR*, 407–414. IEEE.

Mairal, J.; Bach, F.; Ponce, J.; Sapiro, G.; and Zisserman, A. 2008. Discriminative learned dictionaries for local image analysis. In *CVPR*. IEEE.

Raina, R.; Battle, A.; Lee, H.; Packer, B.; and Ng, A. Y. 2007. Self-taught learning: transfer learning from unlabeled data. In *ICML*, 759–766. ACM.

Szlam, A.; Gregor, K.; and LeCun, Y. 2012. Fast approximations to structured sparse coding and applications to object classification. In *ECCV*. Springer. 200–213.

Tierney, S.; Gao, J.; and Guo, Y. 2014. Subspace clustering for sequential data. In *CVPR*, 1019–1026.

Tropp, J. A.; Gilbert, A. C.; and Strauss, M. J. 2006. Algorithms for simultaneous sparse approximation. part i: Greedy pursuit. *Signal Processing* 86(3):572–588.

Wang, J.; Yang, J.; Yu, K.; Lv, F.; Huang, T.; and Gong, Y. 2010. Locality-constrained linear coding for image classification. In *CVPR*, 3360–3367. IEEE.

Wang, F.; Lee, N.; Sun, J.; Hu, J.; and Ebadollahi, S. 2011. Automatic group sparse coding. In *AAAI*.

Wright, J.; Yang, A. Y.; Ganesh, A.; Sastry, S. S.; and Ma, Y. 2009. Robust face recognition via sparse representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 31(2):210–227.

Yang, J.; Yu, K.; Gong, Y.; and Huang, T. 2009. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, 1794–1801. IEEE.

Yang, M.; Zhang, D.; and Feng, X. 2011. Fisher discrimination dictionary learning for sparse representation. In *ICCV*, 543–550. IEEE.

Zhang, Q., and Li, B. 2010. Discriminative k-svd for dictionary learning in face recognition. In *CVPR*, 2691–2698. IEEE.

Zhang, X.; Yu, Y.; White, M.; Huang, R.; and Schuurmans, D. 2011. Convex sparse coding, subspace learning, and semi-supervised extensions. In *AAAI*.

Zhang, G.; Jiang, Z.; and Davis, L. S. 2013. Online semi-supervised discriminative dictionary learning for sparse representation. In *ACCV*. Springer. 259–273.

Zheng, M.; Bu, J.; Chen, C.; Wang, C.; Zhang, L.; Qiu, G.; and Cai, D. 2011. Graph regularized sparse coding for image representation. *Image Processing, IEEE Transactions on* 20(5):1327–1336.