A Multi-task Learning Strategy for Unsupervised Clustering via Explicitly Separating the Commonality

Shu Kong, Donghui Wang
Dept. of Computer Science and Technology, Zhejiang University, Hangzhou 310027, PR China
{*aimerykong, dhwang*}*@zju.edu.cn*

Abstract

In this paper, we propose an unsupervised cluster method via a multi-task learning strategy, called ***Mt-Cluster****. Our MtCluster learns a cluster-specific dictionary for each cluster to represent its sample signals and a shared common pattern pool (the commonality) for the essentially complemental representation. By treating learning the cluster-specific dictionary as a single task, MtCluster works in a multi-task learning manner, in which all the tasks are connected by simultaneously learning the commonality. Actually, the learned cluster-specific dictionary spans the feature space of the corresponding cluster, and the commonality is just used for necessary complemental representation. To evaluate our method, we perform several experiment on public available datasets, and the promising results demonstrate the effectiveness of MtCluster.*

1. Introduction

Clustering is the task of assigning a set of samples into several clusters, such that the samples belonging to the same cluster are more similar to each other than to those in other clusters. It is a common technique for statistical data analysis in various fields, such as machine learning, information retrieval, pattern recognition, image analysis and so forth. Among various clustering algorithms, the k-means clustering algorithm is one of the most widely-used one for unsupervised partitioning [2]. The classic k-means aims to find a centroid or a mean vector for each one of the C clusters, and partitions a given observation into a specific cluster with the nearest mean. However, when the samples scatter within the cluster, k-means may achieve poor performance, because the vector centroid cannot fully represent the scattering observations within the cluster. Therefore, some new representations of the cluster centroid ought to be developed rather than the single vector centroid.

Dictionary learning (DL), as a particular sparse signal model, has risen to prominence in recent years. It aims to learn a (overcomplete) dictionary in which only a few atoms can be linearly combined to well approximate a given signal. DL-based methods have achieved state-of-the-art performances in many application fields, including image denoising [3] and image classification [6]. Recently, a new unsupervised clustering method is developed by learning a dictionary for each cluster [9]. The learned dictionary can well represent the sample signals belonging the a specific cluster in a sparse manner. Therefore, it performs much better than k-means in clustering task.

However, some atoms of some learned dictionaries can be very similar or coherent, thus they can be used for representing signals from different clusters. Owning to this, the clustering performance will be degraded. This problem has been addressed in [8] by adding an incoherence penalty term. But still, it can be further improved in some ways. Empirically, we observe that different classes of signals (*e.g.* images) always share some common patterns, and these shared patterns are not conducive to discrimination of them but are essential for representing them. Motivated by this observation, we propose to separate these common patterns and learn the most discriminative cluster-specific features to achieve better clustering performance. By treating learning the dictionary for each cluster as a single task, we propose a novel method to learn the cluster-specific dictionaries for each cluster in a multi-task learning manner, in which all the tasks are connected by explicitly and simultaneously learning a shared pattern pool (the commonality). We call the proposed method **Mt-Cluster**, and evaluate it through several experiments.

2 Motivation and related works

We start our work by first listing some notations. Suppose there are C clusters and $\mathbf{X} \in \mathbb{R}^{d\times N}$ is the dataset, in which a column is a sample datum $\mathbf{x}_i \in \mathbb{R}^d$.

We use a column vector $\mathbf{y}_i \in \{0,1\}^C$ as the assignment denotation to specify the cluster that $\mathbf{x}_i$ belongs to, that is to say the c^{th} element $\mathbf{y}_i[c] = 1$ if $\mathbf{x}_i$ belongs to the c^{th} cluster and 0 otherwise. As well, let $\mathcal{I}_c$ index all the data belonging to the c^{th} cluster.

k-means. The k-means algorithm is a widely-used approach for clustering and is closely related to our MtCluster. It aims to minimize the following objective over $\mathbf{M} = [\mathbf{m}_c] \in \mathbb{R}^{d\times C}$ and $\mathbf{Y} = [\mathbf{y}_i] \in \mathbb{R}^{C\times N}$:

$$\begin{aligned} f_k =& \sum_{c=1}^{C}\sum_{i\in\mathcal{I}_c} \|\mathbf{x}_i - \mathbf{m}_c\|_F^2 = \|\mathbf{X} - \mathbf{M}\mathbf{Y}\|_F^2 \\ \text{s.t. } & \mathbf{Y} \in \{0,1\}^{C\times N} \text{ and } \mathbf{y}_i^T\mathbf{1} = 1, \text{ for } i = 1,\ldots,N. \end{aligned} \tag{1}$$

When the samples scatter within a specific cluster, the k-means method may achieve unsatisfactory performance, because the centroid cannot fully represent the real signals of this cluster. Therefore, some new representation method need to be developed.

Cluster-specific dictionary as the centroid. In [9], a new method which is based on dictionary learning and sparse coding is proposed for unsupervised clustering, and achieve very promising performance in several clustering tasks. This method aims to learn a dictionary for each cluster, rather than the vector centroid. The cluster-specific dictionary is assumed to be representative for the samples belonging to the corresponding cluster, *i.e.* a few bases of the dictionary can be linearly combined to well represent the samples in a sparse manner. However, two inherent problems of this method are overlooked. One is that the learned dictionaries can share some common bases which can be used for representing signals from different clusters. The other one is the fact that data from different clusters always share some common patterns which do not contribute to discrimination of them and even degrade the performance. The first problem is taken into consideration in [8] by adding a incoherence penalty term among the dictionaries. But we can go beyond this simple modification. Our proposed MtCluster can address both of the problems by learning the most compact and discriminative cluster-specific dictionaries and explicitly separating the common patterns (the commonality). We assume each dictionary spans the feature space of the corresponding cluster and the commonality is just used for the complemental representation. Thus the dictionary do not necessarily need to be overcomplete, and we can simply solve a square square problem for to encode the signals over the dictionary. If we treat learning the dictionary for each cluster of sample signals is a task, then MtCluster is working in a multi-task manner, in which the tasks are connected by simultaneously learning the commonality. Next section presents our MtCluster method.

3 The proposed MtCluster

In the real world, all the samples share some common patterns which do not contribute to discrimination but are essential for reconstruction in DL. For this reason, we propose to separate the shared patterns by explicitly learning a shared feature pool (the commonality) $\mathbf{D}_0 \in \mathbb{R}^{d\times K_0}$ which provides the common feature bases among all the categories to improve discrimination performance. Denote the overall dictionary as $\mathbf{D} = [\mathbf{D}_1,\ldots,\mathbf{D}_c,\ldots,\mathbf{D}_C] \in \mathbb{R}^{d\times K}$, in which $K = kC$ and $\mathbf{D}_c \in \mathbb{R}^{d\times k}$ stands for the c^{th} sub-dictionary, spanning the feature subspace of the c^{th} cluster. Ideally, the class-specific sub-dictionaries should be incoherent with each other to guarantee each sub-dictionary preserve its corresponding feature space. On the other hand, the commonality should be incoherent with all the sub-dictionaries either. Therefore, we adopt an incoherence-penalty term to achieve this, by adding $\|\mathbf{D}^T\mathbf{D}\|_F^2$ and $\|\mathbf{D}^T\mathbf{D}_0\|_F^2$. It is worth noting that by adding the two penalty terms, not only the incoherence among the sub-dictionaries can be achieved to a large extent, but also the bases within a single sub-dictionary is driven to be incoherent by which way the cluster-specific dictionaries can be more discriminative and more compact. Then our goal is to minimize the following objective function f over $\mathbf{D}$, $\mathbf{D}_0$, $\mathbf{a}_i$ and $\mathbf{a}_i^0$:

$$\begin{aligned} f = \sum_{i=1}^{N}\big\{&\|\mathbf{x}_i - \mathbf{D}(\mathbf{I}\otimes\mathbf{y}_i)\mathbf{a}_i - \mathbf{D}_0\mathbf{a}_i^0\|_F^2 + \lambda_1\|\mathbf{a}_i\|_2^2 \\ &+\lambda_2\|\mathbf{a}_i^0\|_1\big\} + \lambda_3\|\mathbf{D}^T\mathbf{D}\|_F^2 + \lambda_4\|\mathbf{D}^T\mathbf{D}_0\|_F^2, \\ &\text{s.t. } \mathbf{y}_i \in \{0,1\}^C, \mathbf{y}_i^T\mathbf{1} = 1, \text{ for } i = 1,\ldots,N, \\ &\{\|\mathbf{d}_j^0\|_2 = 1\}_{j=1}^{K_0} \text{ and } \{\|\mathbf{d}_i\|_1 = 1\}_{i=1}^{K}. \end{aligned} \tag{2}$$

For convenience, we set $\lambda_3 = \lambda_4 = \lambda_D$ in our work. From the objective function, we can see that if we set $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = 0$, $\mathbf{I} = \mathbf{a}_i = 1$, $\mathbf{D} \in \mathbb{R}^{d\times C}$ and eliminate $\mathbf{D}_0$, then f_{obj} can degenerate into k-means algorithm in Eq. 1. Therefore, MtCluster can be cast as a generalized version of k-means.

4 Optimization

In this section, we present the optimization of the objective Eq. 2. In our work, we use an alternating process to update the cluster-specific dictionaries and the commonality, along with the assignment $\mathbf{y}_i$ and the coefficients $\mathbf{a}_i$ and $\mathbf{a}_i^0$.

4.1 Fixing $\mathbf{a}_i$ and $\mathbf{a}_i^0$ to update $\mathbf{D}$ and $\mathbf{D}_0$

Given $\mathbf{D}$ to update the commonality $\mathbf{D}_0$. We propose to update $\mathbf{D}_0 = [\mathbf{d}_1^0, \ldots, \mathbf{d}_{K_0}^0]$ atom by atom, *i.e.* updating atom $\mathbf{d}_j^0$ while fixing all the others. Denote $\tilde{\mathbf{x}}_i = \mathbf{x}_i - \mathbf{D}\mathbf{a}_i$, $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_i] \in \mathbb{R}^{d\times N}$, $\mathbf{A}_0 = [\mathbf{a}_i^0] \in \mathbb{R}^{K_0\times N}$, and $\mathbf{a}_{(j)}^0 \in \mathbb{R}^{1\times N}$ is the j^{th} row of $\mathbf{A}_0$. By ignoring the terms independent of $\mathbf{D}_0$, we have:

$$\mathbf{D}_0 = \underset{\mathbf{D}_0}{\operatorname{argmin}} \|\tilde{\mathbf{X}} - \mathbf{D}_0\mathbf{A}_0\|_F^2 + \lambda_D\|\mathbf{D}_0^T\mathbf{D}\|_F^2. \quad (3)$$

Then, by denoting $\bar{\mathbf{X}} = \tilde{\mathbf{X}} - \sum_{i\neq j}\mathbf{d}_i^0\mathbf{a}_{(i)}^0$, we can update the j^{th} atom of $\mathbf{D}_0$ as below:

$$\mathbf{d}_j^0 = \underset{\mathbf{d}_j^0}{\operatorname{argmin}} \|\bar{\mathbf{X}} - \mathbf{d}_j^0\mathbf{a}_{(j)}^0\|_F^2 + \lambda_D\|\mathbf{D}^T\mathbf{d}_j^0\|_F^2 \quad (4)$$

Through some simple derivations, we can update $\mathbf{d}_j^0$ as below:

$$\mathbf{d}_j^0 = (\|\mathbf{a}_{(j)}^0\|_2^2\mathbf{I} + \lambda_D\mathbf{D}\mathbf{D}^T)^{-1}\bar{\mathbf{X}}{\mathbf{a}_{(j)}^0}^T, \quad (5)$$

Note that $\mathbf{d}_j^0$ should have unit length, *i.e.* $\|\mathbf{d}_j^0\|_2 = 1$. Therefore, the updated atom $\hat{\mathbf{d}}_j^0 = \mathbf{d}_j^0/\|\mathbf{d}_j^0\|_2$, as well, the corresponding row in $\mathbf{A}_0$ should be scaled by $\|\mathbf{d}_j^0\|_2$, *i.e.* $\hat{\mathbf{a}}_{(j)}^0 = \|\mathbf{d}_j^0\|_2\mathbf{a}_{(j)}^0$.

Given $\mathbf{D}_0$ to update the cluster-specific dictionary $\mathbf{D}$. Denote $\grave{\mathbf{x}}_i = \mathbf{x}_i - \mathbf{D}_0\mathbf{a}_i^0$, $\grave{\mathbf{a}}_i = (\mathbf{I}\otimes\mathbf{y}_i)\mathbf{a}_i$, and $\grave{\mathbf{A}} = [\grave{\mathbf{a}}_i] \in \mathbb{R}^{K\times N}$, in which $\mathbf{a}_j$ is the j^{th} row of $\grave{\mathbf{A}}$. Ignore the unrelated terms, then we update the cluster-specific dictionary as below:

$$\mathbf{D} = \underset{\mathbf{D}}{\operatorname{argmin}} \|\grave{\mathbf{X}} - \mathbf{D}\grave{\mathbf{A}}\|_F^2 + \lambda_D\|\mathbf{D}^T\mathbf{D}\|_F^2 + \lambda_D\|\mathbf{D}_0^T\mathbf{D}\|_F^2$$

By denoting $\ddot{\mathbf{X}} = \grave{\mathbf{X}} - \sum_{i\neq j}\mathbf{d}_i\mathbf{a}_{(i)}$ and $h(\mathbf{d}_j) = \|\ddot{\mathbf{X}} - \mathbf{d}_j\mathbf{a}_{(j)}\|_F^2 + \lambda_D(\|\mathbf{d}_j^T\mathbf{D}_0\|_F^2 + \mathbf{d}_j^T\mathbf{d}_j + 2\sum_{i\neq j}\mathbf{d}_j^T\mathbf{d}_i)$, we can update the j^{th} atom of $\mathbf{D}$ by minimizing $h(\mathbf{d}_j)$ w.r.t $\mathbf{d}_j$. Similarly, let $\partial h(\mathbf{d}_j)/\partial\mathbf{d}_j = 0$, we obtain:

$$\mathbf{d}_j = (\frac{\|\mathbf{a}_{(j)}\|_2^2 + \lambda_D}{\lambda_D}\mathbf{I} + \mathbf{D}_0^T\mathbf{D}_0)^{-1}(\frac{1}{\lambda_D}\ddot{\mathbf{X}}\mathbf{a}_{(j)}^T - \sum_{i\neq j}\mathbf{d}_i). \quad (6)$$

As well, the updated atom ought to have unit length, *i.e.* $\hat{\mathbf{d}}_j = \mathbf{d}_j/\|\mathbf{d}_j\|_2$ and corresponding row of $\mathbf{A}$ need to be scaled $\|\mathbf{d}_j\|_2$ times, *i.e.* $\hat{\mathbf{a}}_{(j)} = \|\mathbf{d}_j\|_2\mathbf{a}_{(j)}$.

4.2 Updating the assignment $\mathbf{y}_i$

Note there is a constraint $\mathbf{y}_i \in \{0,1\}^C$ and $\mathbf{y}_i^T\mathbf{1} = 1$ on parameter $\mathbf{y}_i$, but we can divide the objective into C sub-problem to simplify the problem. In other words, we select the cluster of a signal as $\mathbf{D}_c$ which can bring out smallest representation error, *i.e.* $c = \operatorname{argmin}_c \|\mathbf{x}_i - \mathbf{D}_c\mathbf{a}_i - \mathbf{D}_0\mathbf{a}_i^0\|_2^2 + \lambda_1\|\mathbf{a}_i\|_2^2 + \lambda_2\|\mathbf{a}_i^0\|_1$. Then we set $\mathbf{y}_i[c] = 1$ and zeros otherwise.

4.3 Fixing $\mathbf{D}$ and $\mathbf{D}_0$ to update $\mathbf{a}_i$ and $\mathbf{a}_i^0$

Even though there are two penalty norms on the coefficients, we will show it is very easy to obtain the optimal $\mathbf{a}_i$ and $\mathbf{a}_i^0$. If $\mathbf{a}_i^0$ is known, by denoting $\check{\mathbf{x}}_i = \mathbf{x}_i - \mathbf{D}_0\mathbf{a}_i^0$ we can obtain the optimal $\mathbf{a}_i$ in a closed form:

$$\begin{aligned}\mathbf{a}_i &= \underset{\mathbf{a}_i}{\operatorname{argmin}} \|\check{\mathbf{x}}_i - \mathbf{D}(\mathbf{I}\otimes\mathbf{y}_i)\mathbf{a}_i\|_F^2 + \lambda_1\|\mathbf{a}_i\|_2^2 \\ &= (\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\check{\mathbf{x}}_i,\end{aligned} \quad (7)$$

where $\mathbf{Q}_i = \mathbf{D}(\mathbf{I}\otimes\mathbf{y}_i)$. Substituting $\mathbf{a}_i$ back into the objective Eq. 2, we can obtain $\mathbf{a}_i^0$ by minimizing the following:

$$\begin{aligned}&\min_{\mathbf{a}_i^0}\|\mathbf{x}_i - \mathbf{Q}_i(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T(\mathbf{x}_i - \mathbf{D}_0\mathbf{a}_i^0) - \mathbf{D}_0\mathbf{a}_i^0\|_F^2 \\ &+\lambda_1\|(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T(\mathbf{x}_i - \mathbf{D}_0\mathbf{a}_i^0)\|_2^2 + \lambda_2\|\mathbf{a}_i^0\|_1.\end{aligned}$$

Denoting $\mathbf{x}_i^{'} = \mathbf{x}_i - \mathbf{Q}_i(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{x}_i$, $\mathbf{D}^{'} = \mathbf{D}_0 - \mathbf{Q}_i(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{D}_0$, $\mathbf{z}_i = (\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{x}_i$ and $\mathbf{G}_i = (\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{D}_0$, we have the following:

$$\begin{aligned}&\min_{\mathbf{a}_i^0}\|\mathbf{x}_i^{'} - \mathbf{D}^{'}\mathbf{a}_i^0\|_2^2 + \lambda_1\|(\mathbf{z}_i - \mathbf{G}_i\mathbf{a}_i^0)\|_2^2 + \lambda_2\|\mathbf{a}_i^0\|_1 \\ &= \min_{\mathbf{a}_i^0}\left\|\begin{bmatrix}\mathbf{x}_i^{'} \\ \sqrt{\lambda_1}\mathbf{z}_i\end{bmatrix} - \begin{bmatrix}\mathbf{D}^{'} \\ \sqrt{\lambda_1}\mathbf{G}_i\end{bmatrix}\mathbf{a}_j^0\right\|_2^2 + \lambda_2\|\mathbf{a}_i^0\|_1.\end{aligned} \quad (8)$$

Then, we can simply adopt the feature-sign search algorithm [5] solve the desired $\mathbf{a}_j^0$. The overall algorithm is summarized in Algorithm 1. Note that the value objective function decreases at each iteration, and therefore the algorithm converges.

5 Clustering results

Initialization and parameters. To initialize the cluster-specific dictionaries, we first perform k-means on the original data to derive C clusters, then we adopt K-SVD algorithm [1] to initialize the dictionaries for each cluster. Throughout this paper, we set $\lambda_1 = 0.01$, $\lambda_2 = 0.1$, and $\lambda_3 = \lambda_4 = 1$ (more choice of parameters, as well as more experiments, will be discussed in a longer version of this paper). We use the k-means as the baseline, and DLSI method in [8] is also compared with our MtCluster.

Algorithm 1 MtCluster algorithm

Require: training dataset $\mathbf{X}$, the desired size K_0 and k for $\mathbf{D}_0$ and $\mathbf{D}_c$'s respectively, and λ_1, λ_2 and λ_D;

initialize $\mathbf{D}_c$ for $c = 0, 1, \ldots, C$;
while not converge **do**
  update all the coefficients $\mathbf{a}_i^0$ and $\mathbf{a}_i$ by solving Eq. 8 and Eq. 7, respectively;
  update the assign denotation vector $\mathbf{y}_i$ in Section 4.2;
  update the atoms in $\mathbf{D}$ through Eq. 6 with unitization process and scale the corresponding coefficient row;
  update the atoms in $\mathbf{D}_0$ through Eq. 5 with unitization process and scale the corresponding coefficient row;
end while
return the learned $\mathbf{y}_i$'s, $\mathbf{D}_c$'s and $\mathbf{D}_0$.

Table 1. Comparison of error rates (%).

Dataset	k-means	DLSI	**MtCluster**
MNIST	21.2	3.0	**2.4**
USPS	22.3	2.0	**1.3**
ISOLET	20.0	1.5	**1.1**

Dataset and results. The digit from 0 to 5 (C = 6) in the testing set of MNIST [4] and in the training set of USPS[1] are used for evaluation of clustering performance. We also use the last six letters in both training and testing set of ISOLET (C=6) for clustering. For MNIST and USPS, the cluster-specific dictionary of 20 atoms and the commonality of 30 atoms are learned, overall $6 \times 20 + 30 = 150$ which is no greater or much smaller than 360 and 150 for USPS and MNIST respectively in [8]. For ISOLET, 10 atoms for each cluster-specific dictionary and 30 atoms for the commonality are learned, overall $6 \times 10 + 30 = 90$ which equals that in [8]. Direct comparisons are presented in Table 1, from which we can see our MtCluster achieve the best performance.

We also perform our MtCluster for texture segmentation problem [7]. Overlapped 16×16 patches are extracted from the original images and used as the input to our method. Figure. 1 shows the results on one mosaic image. We can see our MtCluster also outperforms DLSI and the baseline k-means in this segmentation task.

From these experiments, we can see the advantage of explicitly separating the commonality for clustering. In other words, the proposed multi-task learning strategy indeed boosts the clustering performance.

6 Conclusion with remarks

A novel clustering method called MtCluster is proposed in this paper, which is based on dictionary learning and works in a multi-task learning manner. MtCluster can be cast as a generalized k-means approach. By treating learning a discriminative and compact cluster-specific dictionary for each cluster as one single task, the tasks are connected by simultaneously and explicitly learning a commonality, which stands for the shared common features among different classes of data. From experiments, we can see the effectiveness of MtCluster.

[1] http://www-i6.informatik.rwth-aachen.de/∼keysers/usps.html

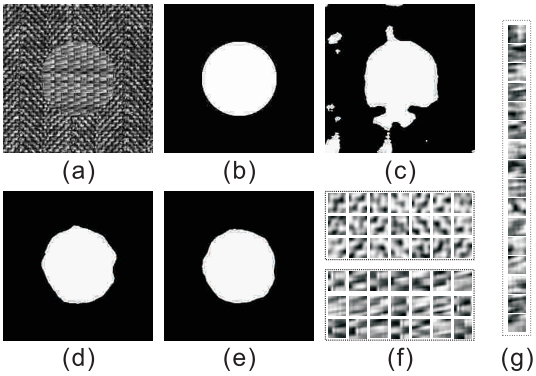

Figure 1. Results on texture segmentation. From (a) to (g): the mosaic image; the ground-truth segmentation; segmentation by k-means; segmentation by [8]; segmentation by MtCluster; the cluster-specific dictionaries learned by MtCluster; the commonality learned by MtCluster. (c)-(e) have 6.92%, 1.42% and 1.04% misclassified pixels respectively.

References

[1] M. Aharon, M. Elad, and A. M. Bruckstein. K-svd: an algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. on Signal Processing*, 2006.

[2] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, New York, 1973.

[3] M. Élad and M. Aharon. image denoising via learned dictionaries and sparse representation. *CVPR*, 2006.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. of IEEE*, 1998.

[5] H. Lee, A. Battle, R. Raina, and A. Y. Ng. efficient sparse coding algorithms. *NIPS*, 2006.

[6] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman. Supervised dictionary learning. *CVPR*, 2008.

[7] G. Peyre. sparse modeling of textures. *J. Mathematical Imaging and Vision*, 2009.

[8] I. Ramirez, P. Sprechmann, and G. Sapiro. classification and clustering via dictionary learning with structured incoherence and shared features. *CVPR*, 2010.

[9] P. Sprechmann and G. Sapiro. Dictionary learning and sparse coding for unsupervised clustering. *ICASSP*, 2010.

A Derivation of $\mathbf{d}_j^0$

$$\begin{aligned}\mathbf{d}_j^0 =& \underset{\mathbf{d}_j^0}{\operatorname{argmin}} \|\bar{\mathbf{X}} - \mathbf{d}_j^0\mathbf{a}_{(j)}^0\|_F^2 + \lambda_D\|\mathbf{D}^T\mathbf{d}_j^0\|_F^2 \\ =& \underset{\mathbf{d}_j^0}{\operatorname{argmin}} -2\text{tr}\bar{\mathbf{X}}^T\mathbf{d}_j^0\mathbf{a}_{(j)}^0 + \|\mathbf{a}_{(j)}^0\|_2^2\|\mathbf{d}_j^0\|_2^2 + \\ & \lambda_D{\mathbf{d}_j^0}^T\mathbf{D}\mathbf{D}^T\mathbf{d}_j^0\end{aligned}$$

Denote $g(\mathbf{d}_j^0) = -2\text{tr}\bar{\mathbf{X}}^T\mathbf{d}_j^0\mathbf{a}_{(j)}^0 + \|\mathbf{a}_{(j)}^0\|_2^2\|\mathbf{d}_j^0\|_2^2 + \lambda_D{\mathbf{d}_j^0}^T\mathbf{D}\mathbf{D}^T\mathbf{d}_j^0$. Let the first derivative of $g(\mathbf{d}_j^0)$ w.r.t $\mathbf{d}_j^0$, we have:

$$\begin{aligned}\frac{\partial g(\mathbf{d}_j^0)}{\partial \mathbf{d}_j^0} =& -2\bar{\mathbf{X}}{\mathbf{a}_{(j)}^0}^T + 2\|\mathbf{a}_{(j)}^0\|_2^2\mathbf{d}_j^0 + 2\lambda_D\mathbf{D}\mathbf{D}^T\mathbf{d}_j^0 \\ =& -2\bar{\mathbf{X}}{\mathbf{a}_{(j)}^0}^T + 2(\|\mathbf{a}_{(j)}^0\|_2^2\mathbf{I} + \lambda_D\mathbf{D}\mathbf{D}^T)\mathbf{d}_j^0 = 0\end{aligned}$$

Then $\mathbf{d}_j^0 = (\|\mathbf{a}_{(j)}^0\mathbf{I} + \lambda_D\mathbf{D}\mathbf{D}^T)^{-1}\bar{\mathbf{X}}{\mathbf{a}_{(j)}^0}^T$.

B Derivation of $\mathbf{d}_j$

To update the atoms $\mathbf{d}_j$ in $\mathbf{D}$, we can easily see:

$$\begin{aligned}\mathbf{d}_j =& \underset{\mathbf{d}_j}{\operatorname{argmin}}\|\ddot{\mathbf{X}} - \mathbf{d}_j\mathbf{a}_{(j)}\|_F^2 + \\ & \lambda_D(\|\mathbf{d}_j^T\mathbf{D}_0\|_F^2 + \|\mathbf{D}^T\mathbf{D}\|_F^2) \\ =& \underset{\mathbf{d}_j}{\operatorname{argmin}}\|\ddot{\mathbf{X}} - \mathbf{d}_j\mathbf{a}_{(j)}\|_F^2 + \lambda_D(\|\mathbf{d}_j^T\mathbf{D}_0\|_F^2 + \\ & \sum_{m\neq j, n\neq j}\mathbf{d}_n^T\mathbf{d}_m + \mathbf{d}_j^T\mathbf{d}_j + \sum_{m\neq j}\mathbf{d}_j^T\mathbf{d}_m + \sum_{n\neq j}\mathbf{d}_j^T\mathbf{d}_n) \\ =& \underset{\mathbf{d}_j}{\operatorname{argmin}}\|\ddot{\mathbf{X}} - \mathbf{d}_j\mathbf{a}_{(j)}\|_F^2 + \lambda_D(\|\mathbf{d}_j^T\mathbf{D}_0\|_F^2 + \mathbf{d}_j^T\mathbf{d}_j + \\ & 2\sum_{i\neq j}\mathbf{d}_j^T\mathbf{d}_i) \\ =& \underset{\mathbf{d}_j}{\operatorname{argmin}} -2\text{tr}\ddot{\mathbf{X}}\mathbf{a}_{(j)}^T\mathbf{d}_j^T + \|\mathbf{a}_{(j)}\|_2^2\|\mathbf{d}_j\|_2^2 + \\ & \lambda_D(\mathbf{d}_j^T\mathbf{D}_0\mathbf{D}_0^T\mathbf{d}_j + \mathbf{d}_j^T\mathbf{d}_j + 2\sum_{i\neq j}\mathbf{d}_j^T\mathbf{d}_i)\end{aligned}$$

Denote $g(\mathbf{d}_j) = -2\text{tr}\ddot{\mathbf{X}}\mathbf{a}_{(j)}^T\mathbf{d}_j^T + \|\mathbf{a}_{(j)}\|_2^2\|\mathbf{d}_j\|_2^2 + \lambda_D(\mathbf{d}_j^T\mathbf{D}_0\mathbf{D}_0^T\mathbf{d}_j + \mathbf{d}_j^T\mathbf{d}_j + 2\sum_{i\neq j}\mathbf{d}_j^T\mathbf{d}_i)$, we can let the first derivative of $g(\mathbf{d}_j)$ equal zero, *i.e.* $\partial g(\mathbf{d}_j)/\partial\mathbf{d}_j = 0$, then:

$$\begin{aligned}\frac{\partial g(\mathbf{d}_j)}{\partial \mathbf{d}_j} =& -2\ddot{\mathbf{X}}\mathbf{a}_{(j)}^T + 2\|\mathbf{a}_{(j)}\|_2^2\mathbf{d}_j \\ & + \lambda_D(2\mathbf{D}_0\mathbf{D}_0^T\mathbf{d}_j + 2\mathbf{d}_j + 2\sum_{i\neq j}\mathbf{d}_i) \\ =& 2(\|\mathbf{a}_{(j)}\|_2^2\mathbf{I} + \lambda_D\mathbf{I} + \lambda_D\mathbf{D}_0\mathbf{D}_0^T)\mathbf{d}_j \\ & + 2(-\ddot{\mathbf{X}}\mathbf{a}_{(j)}^T + \lambda_D\sum_{i\neq j}\mathbf{d}_i) \\ =& 0\end{aligned}$$

$$\mathbf{d}_j = (\frac{\|\mathbf{a}_{(j)}\|_2^2 + \lambda_D}{\lambda_D}\mathbf{I} + \mathbf{D}_0^T\mathbf{D}_0)^{-1}(\frac{1}{\lambda_D}\ddot{\mathbf{X}}\mathbf{a}_{(j)}^T - \sum_{i\neq j}\mathbf{d}_i).$$

C Derivation of $\mathbf{a}_j^0$

$$\begin{aligned}\mathbf{a}_i^0 =& \underset{\mathbf{a}_i^0}{\operatorname{argmin}} \|\mathbf{x}_i - \mathbf{Q}_i(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T(\mathbf{x}_i - \mathbf{D}_0\mathbf{a}_i^0) \\ & - \mathbf{D}_0\mathbf{a}_i^0\|_F^2 + \lambda_1\|(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T(\mathbf{x}_i - \mathbf{D}_0\mathbf{a}_i^0)\|_2^2 \\ & + \lambda_2\|\mathbf{a}_i^0\|_1.\end{aligned}$$

Denoting $\mathbf{x}_i^{'} = \mathbf{x}_i - \mathbf{Q}_i(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{x}_i$, $\mathbf{D}^{'} = \mathbf{D}_0 - \mathbf{Q}_i(\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{D}_0$, $\mathbf{z}_i = (\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{x}_i$ and $\mathbf{G}_i = (\mathbf{Q}_i^T\mathbf{Q}_i + \lambda_1\mathbf{I})^{-1}\mathbf{Q}_i^T\mathbf{D}_0$, we have the following:

$$\begin{aligned}\mathbf{a}_i^0 =& \underset{\mathbf{a}_i^0}{\operatorname{argmin}} \|\mathbf{x}_i^{'} - \mathbf{D}^{'}\mathbf{a}_i^0\|_2^2 + \lambda_1\|(\mathbf{z}_i - \mathbf{G}_i\mathbf{a}_i^0)\|_2^2 + \lambda_2\|\mathbf{a}_i^0\|_1 \\ =& \underset{\mathbf{a}_i^0}{\operatorname{argmin}}\left\|\begin{pmatrix}\mathbf{x}_i^{'} \\ \sqrt{\lambda_1}\mathbf{z}_i\end{pmatrix} - \begin{pmatrix}\mathbf{D}^{'} \\ \sqrt{\lambda_1}\mathbf{G}_i\end{pmatrix}\mathbf{a}_j^0\right\|_2^2 + \lambda_2\|\mathbf{a}_i^0\|_1.\end{aligned}$$

Then, we can simply adopt the feature-sign search algorithm [5] solve the desired $\mathbf{a}_j^0$.