Local-Structure Adaptive Sparse Subspace Learning: An Iterative Approach to Unsupervised Feature Selection

Nan Zhou\textsuperscript{a,b}, Yangyang Xu\textsuperscript{c}, Hong Cheng\textsuperscript{a}, Jun Fang\textsuperscript{a}, Witold Pedrycz\textsuperscript{b}

\textsuperscript{a}Center for Robotics, School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, China
\textsuperscript{b}Department of Electrical and Computer Engineering, University of Alberta, AB, T6G2R3, Canada
\textsuperscript{c}Department of Combinatorics and Optimization, University of Waterloo, ON, N2L3G1, Canada

Abstract

Subspace learning is becoming more and more popular thanks to its capabilities of good interpretation. However, existing approaches do not adapt both local structure and self reconstruction very well. We propose local-structure adaptive sparse subspace learning (ASSL) model for unsupervised feature selection. In this paper, we formulate the feature selection process as a subspace learning problem and incorporate a regularization term to preserve the local structure of the data. Furthermore, we develop a greedy algorithm to establish the basic model and an iterative strategy based on an accelerated block coordinate descent is used to solve the local-structure ASSL problem. We also provide the global convergence analysis of the proposed ASSL algorithm. Extensive experiments are conducted on real-world datasets to show the superiority of the proposed approach over several state-of-the-art unsupervised feature selection approaches.

Keywords: Machine learning, Feature selection, Subspace learning, Unsupervised learning

1. Introduction

With the advances in data processing, the dimensionality of the data also increases and can be extremely high in many fields such as computer vision, machine learning and image processing. The high dimensionality of the data not only greatly increases the time and storage space required to realize data analysis but also introduces much redundancy and noise which can decrease the accuracy of the ensuing method. Hence, dimensionality reduction is an important and often necessary preprocessing step to facilitate clustering and classification. There are two widely used dimensionality reduction approaches, namely: subspace learning and feature selection. Subspace learning aims to learn a projection which can map the original features into a lower-dimensional subspace by some transformation forming new features \cite{1,2}. Feature selection aims to select a subset of the features following a certain criterion \cite{3}. Both the two approaches can successfully address the problems we mentioned at the beginning of the discussion \cite{4,5}.

Email addresses: nzhouuestc@gmail.com (Nan Zhou), yangyang.xu@waterloo.ca (Yangyang Xu), hcheng@uestc.edu.cn (Hong Cheng), JunFang@uestc.edu.cn (Jun Fang), wpedrycz@ualberta.ca (Witold Pedrycz)

Preprint submitted to Elsevier June 4, 2015
In general, subspace learning methods can be categorized into two classes: supervised and unsupervised subspace learning, depending on whether the subspace learning process is supplied with class labels. The most classic subspace learning method is Principal Component Analysis (PCA) [6] which considers the global structures of the data. Its goal is to find a set of mutually orthogonal basis function which can capture the directions of maximum variance in the data. The local structure always has more discriminative information for specific tasks for both supervised and unsupervised situation, hence many subspace learning methods are proposed by preserving the local structure of the data by graph embedding. Some popular ones include Locality Preserving Projection (LPP) [7], Neighborhood Preserving Embedding [8], Linear Discriminant Analysis (LDA) [9] and Sparsity Preserving Projections [10]. However, all these models only consider the local structure or global structure.

Feature selection methods also can be categorized into supervised and unsupervised ones. The labels of the data contain essential discriminative information which can guide the feature selection process. Hence, the supervised feature selection methods can often efficiently select useful features. Commonly used supervised feature selection methods include Fisher score [11], Pearson correlation coefficient [12], and mutual information [13]. For some data mining tasks, however, we do not have label information, and this calls for unsupervised feature selection. Without label information to guide, unsupervised feature selection methods have to explore the hidden structure in the data. The intrinsic structure information that is often used includes samples similarity [14, 7], local structure [15, 16], and global structure [17, 18].

The blessing of dimensionality tells that the high dimensional data always has a lower dimensional structure that inspires us to pursue a small subset of the possibly overwhelmingly large number of features. In recent years, sparsity regularized methods have been widely used in many fields such as computer vision [19, 20], image processing [21], signal recovery [22] and so on. Group sparsity which is used in multi-task learning [23] and joint representation [24] is also widely used in feature selection tasks [25, 26, 27]. It models the feature selection process as a regression problem and imposes the group sparsity on the weight matrix to select the useful features.

The paper comes with the following contributions:

1. We propose a novel unsupervised subspace learning model, which considers both global reconstruction information and local structure information of the data. Because of the group sparsity regularization term, this model is suitable for feature selection tasks. The model is derived by relaxing an existing combinatorial model with its 0-1 variables relaxed to nonnegative ones, and since the solution of the original model is extremely sparse, it further employs a group sparsity regularization term to promote sparsity of the solution.

2. Due to the combinatorial characteristics of the original model, we, for the first time, propose a greedy algorithm to solve the problem. The relaxed model results in a continuous but nonconvex program. Among various continuous optimization methods, we choose an accelerated block coordinate descent (BCD) method. The BCD method utilizes the bi-convexity structure of the problem and has been found to be efficient for our purposes.

3. We establish a global convergence result of the BCD method for our problem by assuming the existence of a full rank limit point. Because of the peculiarity of the formulated problem, the result is new and not implied by any existing convergence results of BCD.

4. We conduct extensive experimental studies. The proposed methods are tested on six real-world datasets coming from different areas and compared to five state-of-the-art unsupervised feature selection algorithms. The results demonstrate the superiority of our methods.
The paper is organized as follows. In Sect. 2, we give a brief review of recent related studies on feature selection. Sect. 3 reviews two local structure preservation methods and presents a local structure preserved sparse subspace learning model. In Sect. 4, we present an algorithms leading to the solution of the problem. Convergence results are also shown. Experimental results are reported in Sect. 5. Finally, we conclude this paper in Sect. 6.

2. Related Studies

The well-known subspace learning method is PCA [6], which maximizes the global data structure information in the principal space and hence it is optimal for data reconstruction. Data reconstruction capturing the discriminative information plays a crucial role in pattern recognition [28]. Local structure always contains important discriminative information [29]. Therefore, many subspace learning methods preserve different local structures of the data for different problems and can get better performance than traditional PCA. Many of these methods use the linear extension of graph embedding (LGE) methods to preserve the local structure. With different choices of the graph adjoint matrix, LGE framework will lead to many subspace learning methods. Some popular ones include Linear Discriminant Analysis (LDA) [6], Locality Preserving Projection (LPP) [7] and Neighborhood Preserving Embedding (NPE) [8]. LDA is a supervised subspace learning method. It determines the linear projection which can maximally separate the data between different classes and clusters the data in the same class. LPP can be used in both supervised and unsupervised scenarios by adding or without the label information in the graph. It is used to find the projection which can preserve the nearest neighbors information. NPE is suitable for both supervised and unsupervised scenarios. It finds the linear projection which can best preserve the nearest neighbors’ reconstruction property. However, all these locality preservation methods involve dense matrices eigen-decomposition which is very expensive in both time and memory. Considering the drawback, Cai et al. [30] proposed a two-step Spectral Regression (SR) method to transform the eigen-decomposition problem into two steps regression problem which is very efficient and flexible to add the regularization term in the regression step. However, this method only considers the local structure.

The blessing of dimensionality tells us the high dimensional data always has the redundancy, hence sparsity as a useful tool to reduce the redundancy has been widely used in many areas, such as computer vision, data processing and machine learning. Considering the redundancy of the subspace learning, Zou et al. [31] proposed an elegant sparse PCA method (SPCA), which transforms the traditional PCA problem into regression problem and uses the “Elastic Net” framework to solve $\ell_1$ regularization problem. However, sometimes, the samples in the subspace learning might be outliers, especially in the problem of background modeling or face recognition. To overcome this problem, Wang et al. [2] proposed a robust elastic net model (REN) which adds the M-estimator in the least square term of the SPCA model and improves the robustness of the SPCA. Both the SPCA and REN only consider the reconstruction information for the learned subspace. Considering the locality preservation, Moghaddam et al. [32] proposed a spectral bounds framework for sparse subspace learning. However, the greedy algorithm they proposed has very high computation complexity in backward elimination step, hence it is not suitable for high dimensional data. Cai et al. [33] proposed a unified sparse subspace learning method
based on spectral regression model, which adds the $\ell_1$ regularization term in the regression step. However, all these sparse subspace learning methods only consider one type of information: local information or global reconstruction information.

Because of the connections between subspace learning and feature selection, recently, some feature selection methods that combine the feature selection process with subspace learning have been proposed. Cai et al. [5] combined the sparse subspace learning with feature selection and proposed the Multi-Cluster Feature Selection (MCFS) method. In the feature selection process, the transformation matrix can be seen as the weight matrix for feature selection. However, MCFS method uses the $\ell_1$ regularization term to control the weight matrix. However, as mentioned by Gu et al. in [4], $\ell_1$ term is not quite suitable for feature selection, and they used the $\ell_{2,1}$ regularization term in the regression step of the subspace learning process to control the weight matrix and improve the feature selection result. However, both of these methods only use the local structure information in subspace learning. Considering the global reconstruction information, Wang et al. [34] proposed an unsupervised feature selection framework, which used the global reconstruction information in subspace learning and used the orthogonality to constrain the weight matrix for a specific feature selection task. However, in general, such a requirement cannot be satisfied since feature weight vectors are not necessarily orthogonal with each other in practice. In addition, it only considers the reconstruction information and does not take local structure of the data into consideration. In practical application, the local structure often contains essential discriminative information as demonstrated in [29]. Instead of enforcing the feature weight matrix to be orthogonal, our model uses a regularization term to encourage row group sparsity, which is more reasonable [4] and in addition, it includes a local structure preserving term to adapt the local structure of the data.

To facilitate the presentation of the material, Table 1 contains the notation used in this study.

| Notation | Description |
|----------|-------------|
| $n$      | The number of instances |
| $d$      | The number of features |
| $A_i$    | The $i^{th}$ row of the matrix $A$ |
| $K$      | The number of selected features |
| $m$      | The number of nearest neighbors |
| $\|W\|_{2,1}$ | $\sum_i \|W_i\|_2$, the sum of the $\ell_2$-norm of rows in $W$ |
| $\|x\|_0$ | $\#\{x_i \neq 0\}$, the number of nonzero elements in vector $x$ |
| $|I|$    | cardinality of set $I$ |

3. The Proposed Framework of Local Structure Adaptive Sparse Subspace Learning

In this section, we introduce our feature selection models that encourage global data fitting and also preserve local structure information of the data. The first model is of combinatorial nature, only allowing 0-1 valued variables. The modelling idea is intuitive, but it is not easy to find a good approximate solution to the problem. The second model relaxes the first one and becomes its continuous counterpart. Various optimization methods can be utilized to determine its solution. More importantly, we find that the relaxed model can most times give better practical performance than the original one; refer the numerical results reported in Section 5.
3.1. A Generic Formulation

Given $n$ data samples $\{p_i\}_{i=1}^n$, located in the $d$-dimensional space, the objective of feature selection is to find a small set of features that can capture most useful information of the data and better serve for classification or clustering purpose. One natural way to measure the information content is to see how close the original data samples are to the learned subspace spanned by the selected features. Mathematically, the distance of a vector $x$ to a subspace $X$ can be represented as $\|x - \mathcal{P}_X(x)\|_2$, where $\mathcal{P}_X$ denotes the projection onto $X$ and $\| \cdot \|_2$ is the Euclidean 2-norm. Hence, the feature selection problem can be described as follows

$$
\min_{W,H} \frac{1}{2} \|X - XWH\|_F^2 \\
\text{s.t.} \quad W \in \{0, 1\}^{d \times K}, \quad W^\top 1_{d \times 1} = 1_{K \times 1}, \quad \|W1_{K \times 1}\|_0 = K.
$$

(1)

where $X = [p_1, p_2, \ldots, p_n]^\top \in \mathbb{R}^{n \times d}$. Concerning the proposed model, we make a few remarks:

1. Given $W$, the optimal $H$ produces the coefficients of all $d$ features projected on the subspace spanned by the selected features. Hence, the (1) expresses the distance of $X$ to the learned subspace.
2. The matrix $W$ is the selection matrix with the entries of “0” or “1”. The constraint $W^\top 1_{d \times 1} = 1_{K \times 1}$ enforces that each column of $W$ has only one “1”. Therefore, at most $K$ features are selected.
3. The constraint $\|W1_{K \times 1}\|_0 = K$ guarantees that no feature will be selected more than once, and thus exactly $K$ features will be chosen.

The recent work [34] mentions to use the 0-1 feature selection matrix, but it does not explicitly formulate an optimization model like the (1). As shown in [29], a local structure of the data can often include important discriminative information to distinguish different samples. To make the learned subspace preserve local structure, one can add a regularization term to the objective to promote such structural information, namely, to solve the regularized model

$$
\min_{W,H} \frac{1}{2} \|X - XWH\|_F^2 + \mu \text{Loc}(W) \\
\text{s.t.} \quad W \in \{0, 1\}^{d \times K}, \quad W^\top 1_{d \times 1} = 1_{K \times 1}, \quad \|W1_{K \times 1}\|_0 = K.
$$

(2)

where $\text{Loc}(W)$ is a local structure promoting regularization term, and $\mu$ is a parameter to balance the data fitting and regularization. In the next subsection, we introduce different forms of $\text{Loc}(W)$.

3.2. Local Structure Preservation Methods

Local structure of the data often contain important information that can be used to distinguish the samples [5, 7]. A predictor utilizing local structure information can be much more efficient than that only using global information [29]. Therefore, one may anticipate that the learned lower dimensional subspace could preserve local structure of the training data. We briefly review two widely used local structure preservation methods.
3.2.1. Local Linear Embedding

The Local Linear Embedding (LLE) \cite{15} method first finds the set $N_m(p_j)$ of $m$ nearest neighbors for all $j$ and then constructs the similarity matrix $S$ as the (normalized) solution of the following problem

$$\min_S \sum_{i=1}^{n} \| p_i - \sum_{j=1}^{n} S_{ij} p_j \|^2_2,$$

s.t. $S_{ij} = 0, \forall j \notin N_m(p_i), \forall i$. \hfill (3)

One can regard $S_{ij}$ as the coefficient of the $j$-th sample when approximating the $i$-th sample, and the coefficient is zero if the $j$-th sample is not the neighbor of the $i$-th one. After obtaining $S$ from (3), LLE further normalizes it such that $\sum_{j=1}^{n} S_{ij} = 1$. Then it computes the lower-dimensional representation $Y = W^T X^T \in \mathbb{R}^{K \times n}$ through solving the problem

$$\min_W \sum_{i=1}^{n} \sum_{j=1}^{n} S_{ij} \| W^T p_i - W^T p_j \|^2_2.$$

Note that if $W$ is a selection matrix, $W^T p_j$ becomes a lower-dimensional sample, keeping the $K$ selected features by $W$ and removing all other features. Let $L = (I - S)^T (I - S)$, where $I$ is the $n \times n$ identity matrix. Then it is easy to see that (4) can be equivalently expressed as

$$\min_W \text{Tr}(W^T X^T L X W).$$

3.2.2. Linear Preserve Projection

In Linear Preserve Projection \cite{36} (LPP) method, the similarity matrix $S$ is generated by

$$S_{ij} = \begin{cases} \exp\left(\frac{\| p_i - p_j \|^2}{2\sigma^2}\right) & p_i \in N_m(p_j) \text{ or } p_j \in N_m(p_i) \\ 0 & \text{otherwise} \end{cases},$$

where $N_m(p_i)$ is the set of $m$ nearest neighbors of $p_i$. The LPP method requires the lower-dimensional representation to preserve the similarity of the original data and forms the linear transformation $W$ by solving the following optimization problem

$$\min_W \sum_{i,j=1}^{n} S_{ij} \| W^T p_i - W^T p_j \|^2_2.$$

Let $L = D - S$ be the Laplacian matrix, where $D$ is a diagonal matrix, called degree matrix, with diagonal elements $D_{ii} = \sum_{j=1}^{n} S_{ij}, \forall i$. Then (7) can be equivalently expressed as

$$\min_W \text{Tr}(W^T X^T L X W).$$

3.3. Relaxed Formulation

The problem (2) is of combinatorial nature, and we do not have many choices to directly solve it. In the next section, we develop a greedy algorithm, which chooses $K$ features one by one, with each selection decreasing the objective the most among all the remaining features. Although the greedy method can sometimes produce a satisfactory solution, it does not perform reliably. For
this reason, we seek an alternative way to select features by first relaxing (2) to a continuous
problem and then employing a reliable optimization method to solve the relaxed problem. As
observed in our tests, the relaxed method can perform comparably well with and most of times
much better than the original one.

As remarked at the end of Section 3.1, any feasible solution $W$ is nonnegative and has at most $K$
non-zero rows. If $K \ll d$ (that usually is satisfied), then $W$ has lots of zero rows. Based on these
observations, we relax the 0-1 constraint to nonnegativity constraint and the hard constraints
$W^T 1_{d \times 1} = 1_{K \times 1}, \|W 1_{K \times 1}\|_0 = K$ to $g(W) \leq K$, where $g(W)$ measures the row-sparsity of $W$. One
choice of $g(W)$ is group Lasso [37], i.e.,

$$g(W) = \sum_{i=1}^{d} \|W_i\|_2,$$

(9)

where $W_i$ denotes the $i$-th row of $W$. Some other alternatives of $g$ can also be used for promoting
row-sparsity, such as group infinity norm used in [38][39] for multi-class support vector machine.
Hence, we relax (2) to read as follows

$$\min_{W, H} \frac{1}{2} \|X - XWH\|_F^2 + \mu \text{Loc}(W)$$

s.t. $W \geq 0, \ g(W) \leq K,$

(10)

or equivalently

$$\min_{W, H} \frac{1}{2} \|X - XWH\|_F^2 + \mu \text{Loc}(W) + \beta g(W)$$

s.t. $W \geq 0,$

(11)

where $\beta$ is a parameter corresponding to $K$. We focus on (11) only because it is easier to solve
than (10). The drawback of using (11) instead of (10) is that we need to pay more attention to
choose the parameter $\beta$. However, as shown in Section 5 a wide range of values of $\beta$ can give
satisfactory practical performance.

Before completing this section, let us make some remarks on the relaxed model. Originally,
$W$ is restricted to have exactly $K$ non-zeros, so it should be extremely sparse. One can consider
to include a sparsity-promoting term (e.g., $\ell_1$-norm) in the objective of (11). However, doing so
is not necessary since both $g(W)$ and the nonnegativity constraint encourage sparsity of $W$, and
numerically we notice that $W$ output by our algorithm is indeed very sparse. Another point worth
mentioning is that the elements of $W$ given by (11) are real numbers and do not automatically
select $K$ features. For the purpose of feature selection, after obtaining a solution $W$, we choose
the features corresponding to the $K$ rows of $W$ that have the largest norms because larger values
imply more important roles played by the features.

Our model is similar to the Matrix Factorization Feature Selection (MFFS) model proposed in
[34]. The difference is that the MFFS model restricts the matrix $W$ to be orthogonal while we only
encourage the sparsity of $W$. Although orthogonal $W$ makes their model closer to the original
model (1), it increases difficulty of solving their problem. In addition, MFFS does not utilize
local structure preserving term as we do and thus may lose some important local information.
The numerical tests in Sec. 5 demonstrate that the proposed model along with an iterative method
can produce better results than those obtained by using the MFFS method.
4. Solving the Proposed Sparse Subspace Learning

In this section, we present heuristic algorithms to approximately solve (2) and (11). Throughout the rest of the paper, we assume that Loc($W$) takes the function either as (5) or (8) and $g(W)$ is given by (9). Due to the combinatorial structure of (2), we propose a greedy method to solve it. The problem (11) is smooth, and various optimization methods can be applied. Although its objective is nonconvex jointly with respect to $W$ and $H$, it is convex with regard to one of them while the other one is fixed. Based on this property, we choose the block coordinate descent method to solve (11).

4.1. Its Greedy Strategy

In this subsection, a greedy algorithm is developed for selecting $K$ out of $d$ features based on (2). The idea is as follows: each time, we select one from the remaining unselected features such that the objective value is decreased the most. We begin the design of the algorithm by making the following observation.

**Observation 1.** Let $I_1$ and $I_2$ be two index sets of features. Assume $I_1 \subseteq I_2$, and $X_{I_1}$ and $X_{I_2}$ are submatrices of $X$ with columns indexed by $I_1$ and $I_2$ respectively. Then

$$\min_{H_1} \|X - X_{I_1}H_1\|^2_F \geq \min_{H_2} \|X - X_{I_2}H_2\|^2_F.$$  \hspace{1cm} (12)

From the above observation, if the current index set of selected features is $I$, the data fitting will become better if we enlarge $I$ by adding more features. Below we describe in details on how to choose such additional features. Assume $X$ is normalized such that $\|x_j\|_2 = 1$, $j = 1, \ldots, d$, \hspace{1cm} (13)

where $x_j$ denotes the $j$-th column of $X$. Let $I$ be the current index set of selected features. The optimal $H$ to $\min_{H} \|X - X_{I}H\|_F$ is given by

$$H^* = (X_{I}^TX_I)^{+}X_{I}^TX,$$ \hspace{1cm} (14)

where $^{+}$ denotes the Moore-Penrose pseudoinverse of a matrix. Now consider to add one more feature into $I$, say the $j$-th one. Then the lowest data fitting error is

$$\min_{h} \|X - X_{I}H^* - x_jh\|^2_F = \min_{h} \|h\|^2_F - 2\langle h, x_j^T(X - X_{I}H^*) \rangle + \|X - X_{I}H^*\|^2_F$$

$$= -\|x_j^T(X - X_{I}H^*)\|^2_F + \|X - X_{I}H^*\|^2_F,$$

where the last equality is achieved at $h = x_j^T(X - X_{I}H^*)$. Hence, we can choose $j$ such that $\|x_j^T(X - X_{I}H^*)\|_2$ is the largest among all features not present in $I$.

Carrying out a comparison using $\|x_j^T(X - X_{I}H^*)\|_2$, we find that $\|x_j^T(X - X_{I}H^*)\|_1$ can serve better. It turns out that the latter is exactly the correlation between $x_j$ and the residual $X - X_{I}H^*$. Denote the correlation between $x_j$ and $X$ as

$$\text{Cor}(x_j, X) = \sum_{s=1}^{d} |x_j^T x_s|.$$
As shown in [40], if \( \text{Cor}(x_i, X) \) is large, then the columns of \( X \) can be better linearly represented by \( x_i \). To preserve local structure, we need also incorporate \( \text{Loc}(W) \). If the set of selected features is \( I \), then

\[
\text{Loc}(W) = Tr(W^TX^TXW) = \sum_{i \in I} x_i^T L x_i.
\]

Assuming \( L = D - S \), i.e., using the LPP method in section 3.2.2 (that is used throughout our tests), we have from [13] that

\[
\min_{j \not\in I} x_j^T L x_j \Leftrightarrow \max_{j \not\in I} x_j^T S x_j.
\]

Therefore, we can enlarge \( I \) by adding one more feature index \( j^* \) such that

\[
j^* = \arg \max_{j \in \Omega} \text{Cor}(x_j, X - X_I H^*) + x_j^T S x_j,
\]

where \( H^* \) is given in (14), and we have set \( \mu = 1 \) in (2) for simplicity. Algorithm 1 summarizes our greedy method, and for better balancing the correlation and local structure preserving terms, we normalize both of them in the 5th line of Algorithm 1.

**Algorithm 1** Greedy Locally Preserved Subspace Learning (GLPSL)

1. **Input**: Data matrix \( X \in \mathbb{R}^{n \times d} \), the number of features which need to selected \( K \).
2. **Output**: The index of selected features \( I \subseteq \{1, \ldots, d\} \), \( |I| = K \).
3. Initialize residual \( R = X \), candidate set \( \Omega = \{1, 2, \ldots, d\} \), selected set \( I = \emptyset \).
4. **for** \( i = 1 \) to \( K \) **do**
   5. \( i \leftarrow \arg \max_{i \in \Omega} \text{Cor}(x_i, X) + x_i^T S x_i \).
   6. \( \Omega \leftarrow \Omega \setminus \{i\} \) and \( I = I \cup \{i\} \).
   7. \( R \leftarrow X - X_I (X_I^TX_I)^+X_I^TX \).
5. **end for**

4.2. Its Application on Feature Selection

In this subsection, we present an alternative method for feature selection based on (11). Utilizing bi-convexity of the objective, we employ the accelerated block coordinate update (BCU) method proposed in [41] to solve (11). As explained in [41], BCU especially fits to solving bi-convex optimization problems like (11). Compared to traditional optimization methods such as gradient descent and the interior-point method, BCU can have much lower computational complexity and also maintain fast convergence. In addition, it has a guaranteed global sequence convergence when solving (11).

Following the framework of BCU, our algorithm is derived by alternatingly updating \( W \) and \( H \), one at a time while the other one is fixed at its most recent value. Specifically, let

\[
f(W, H) = \frac{1}{2} \|X - XWH\|_F^2 + \frac{H}{2} Tr(W^TX^TXW),
\]

\[
g_{\beta}(W) = \beta \|W\|_{2,1}.
\]
At the $k$-th iteration, we perform the following updates:

$$
W^{k+1} = \arg\min_{W} \langle \nabla_{w} f(W^{k}, H^{k}), W - W^{k} \rangle + \frac{L_{w}^{k}}{2} \|W - W^{k}\|_{2}^{2} + g_{\beta}(W),
$$

(17a)

$$
H^{k+1} = \arg\min_{H} f(W^{k+1}, H),
$$

(17b)

where we take $L_{w}^{k}$ as the Lipschitz constant of $\nabla_{w} f(W, H)$ and

$$
\hat{W}^{k} = W^{k} + \omega_{k}(W^{k} - W^{k-1})
$$

(18)

is an extrapolated point with weight $\omega_{k} \in [0, 1]$, $\forall k$.

Note that the $H$-subproblem (17b) can be simply reduced to a linear equation and has the closed-form solution in the form

$$
H^{k+1} = \left([W^{k+1}]^{\top}X^{\top}X(W^{k+1})\right)^{\top}X^{\top}X,
$$

(19)

where $\dagger$ denotes a Moore-Penrose pseudoinverse. If $H$ is restricted to be nonnegative, in general, (17b) does not exhibit a closed-form solution. In this case, one can update $H$ in the same manner as that of $W$, i.e., completing a block proximal-linearization update. In the following, we discuss in details on how to solve $W$-subproblem (17a) and the parameter settings.

4.2.1. Parameter settings

By direct computation, it is not difficult to see that

$$
\nabla_{w} f(W, H) = X^{\top}(XWH - X)H^{\top} + \mu X^{\top}LXW.
$$

For any $\hat{W}, \tilde{W}$, we have

$$
\|\nabla_{w} f(\hat{W}, H) - \nabla_{w} f(\tilde{W}, H)\|_{F}
= \|X^{\top}(X\tilde{W}H - X)H^{\top} + \mu X^{\top}LX\tilde{W} - X^{\top}(X\hat{W}H - X)H^{\top} - \mu X^{\top}LX\hat{W}\|_{F}
\leq \|X^{\top}(X\tilde{W}H - X)H^{\top} - X^{\top}(X\hat{W}H - X)H^{\top}\|_{F} + \|\mu X^{\top}LX\tilde{W} - \mu X^{\top}LX\hat{W}\|_{F}
= \|X^{\top}X(\hat{W} - \tilde{W})H^{\top}\|_{F} + \|\mu X^{\top}LX(\tilde{W} - \hat{W})\|_{F}
\leq (\|X^{\top}X\|_{2})\|\mu X^{\top}LX\|_{2}\|\hat{W} - \tilde{W}\|_{F},
$$

where $\|A\|_{2}$ denotes the spectral norm and equals the largest singular value of $A$, the first inequality follows from the triangle inequality of norm, and the last inequality is from the fact $\|AB\|_{F} \leq \|A\|_{2}\|B\|_{F}$ for any matrices $A$ and $B$ of appropriate sizes. Hence, $\|X^{\top}X\|_{2}\|HH^{\top}\|_{2} + \|\mu X^{\top}LX\|_{2}$ is a Lipschitz constant of $\nabla_{w} f(W, H)$ with respect to $W$, and in (17a), we set

$$
L_{w}^{k} = \|H^{k}(H^{k})^{\top}\|_{2}\|X^{\top}X\|_{2} + \|\mu X^{\top}LX\|_{2}.
$$

(20)

As suggested by [41], we set the extrapolation weight as

$$
\omega_{k} = \min \left\{ \hat{\omega}_{k}, \delta_{\omega} \sqrt{\frac{L_{w}^{k-1}}{L_{w}^{k}}} \right\},
$$

(21)
where $\delta_0 < 1$ is predetermined and $\hat{\omega}_k = \frac{\hat{\omega}_{k-1}}{t_k}$ with

$$t_0 = 1, \quad t_k = \frac{1}{2} \left( 1 + \sqrt{1 + 4t_{k-1}^2} \right).$$

The weight $\hat{\omega}_k$ has been used to accelerate proximal gradient method for convex optimization problem (cf. [42]). It is demonstrated in [43] that the extrapolation weight in (21) can significantly accelerate BCU for nonconvex problems.

\textbf{Algorithm 2} Proximal operator for nonnegative group Lasso: $W = \text{Prox-NGL}(Y, \lambda)$

\begin{algorithmic}
\FOR{$i = 1, \ldots, d$}
\STATE Let $y$ be the $i$-th row of $Y$ and $I$ the index set of positive components of $y$
\STATE Set $w$ to a zero vector 
\IF{$\|y_I\|_2 > \lambda$}
\STATE Let $w_I = (\|y_I\|_2 - \lambda) \frac{y_I}{\|y_I\|_2}$
\ENDIF
\STATE Set the $i$-th row of $W$ to $w$
\ENDFOR
\end{algorithmic}

4.2.2. Solution of $W$-subproblem

Note that (17a) can be equivalently written as

$$\min_{W \geq 0} \frac{1}{2} \left\| W - \left( \hat{W}^k - \frac{1}{L_u} \nabla_W f(\hat{W}^k, H^k) \right) \right\|_F^2 + \frac{1}{L_u} g_p(W),$$

which can be decomposed into $d$ smaller independent problems, each one involving one row of $W$ and coming in the form

$$\min_{x \geq 0} \frac{1}{2} \left\| x - y \right\|_2^2 + \lambda \|x\|_2.$$  \hfill (22)

We show that (22) has a closed-form solution and thus (17a) can be solved explicitly.

\textbf{Theorem 1.} Given $y$, let $I = \{i : y_i > 0\}$ be the index set of positive components of $y$. Then the solution $x^*$ of (22) is given as follows

1. For any $i \notin I$, $x^*_i = 0$;
2. If $\|y_I\|_2 \leq \lambda$, then $x^*_I = 0$; otherwise, $x^*_I = (\|y_I\|_2 - \lambda) \frac{y_I}{\|y_I\|_2}$.

\textbf{Proof.} For $i \notin I$, we must have $x^*_i = 0$ because if $x^*_i > 0$, setting the $i$-th component to zero and keeping all others unchanged will simultaneously decrease $(x_i - y_i)^2$ and $\|x\|_2$. Hence, we can reduce (22) to the following form

$$\min_{x_i \geq 0} \frac{1}{2} \left\| x_I - y_I \right\|_2^2 + \lambda \|x\|_2.$$  \hfill (23)

Without nonnegativity constraint on $x_I$, the minimizer of (23) is given by item 2 of Theorem 1 (for example, see [44]). Note that the given $x^*_I$ is nonnegative. Hence, it solves (23), and this completes the proof.

The above proof gives a way to find the solution of (22). Using this method, we can explicitly form the solution of (17a) by the subroutine Prox-NGL in Algorithm 2, where $Y \in \mathbb{R}^{d \times k}$ and $\lambda > 0$ are inputs, and $W$ is the output. Arranging the above discussion together, we have the pseudocode see Algorithm 3 for solving (11).
Algorithm 3: Local-Structure Adaptive Sparse Subspace Learning (ASSL)

1: **Input**: Data matrix $X \in \mathbb{R}^{n \times d}$, the number of selected features $K$ and parameter $\beta, \mu$.
2: **Output**: Index set of selected features $I \subseteq \{1, \ldots, d\}$ with $|I| = K$.
3: Initialize $W^0, H^0$, choose a positive number $\delta_{\omega} < 1$; set $k = 1$.

4: while Not convergence do
5:     Compute $L_{kw}^k$ and $\omega_k$ according to (20) and (21) respectively.
6:     Let $\hat{W}^k = W^k + \omega_k(W^k - W^{k-1})$.
7:     Update $W^{k+1} \leftarrow \text{Prox-NGL}(\hat{W}^k - \frac{L_{KW}}{L_{Kw}} \nabla_W f(\hat{W}^k, H^k), \frac{\beta L_{KW}}{L_{Kw}})$.
8:     Update $H^{k+1} \leftarrow (19)$.
9:     if $F(W^{k+1}, H^{k+1}) \geq F(W^k, H^k)$ then
10:        Set $\hat{W}^k = W^k$.
11:     else
12:        Let $k \leftarrow k + 1$.
13:     end if
14: end while
15: Normalize each column of $W$.
16: Sort $\|W_i\|_2, i = 1, \ldots, d$ and select features corresponding to the largest $K$ one.

4.3. Convergence analysis

In this section, we analyze the convergence of Algorithm ASSL. Let us denote

$$\iota_+(W) = \begin{cases} 0, & \text{if } W \geq 0, \\ +\infty, & \text{otherwise} \end{cases}$$

be the indicator function of the nonnegative quadrant. Also, let us denote

$$F(W, H) = f(W, H) + g_\beta(W) + \iota_+(W).$$

Then the problem (11) is equivalent to

$$\min_{W, H} F(W, H),$$

and the first-order optimality condition is $0 \in \partial F(W, H)$. Here, $\partial F$ denotes the subdifferential of $F$ (see [45] for example) and equals $\nabla F$ if $F$ is differentiable and a set otherwise. By Proposition 2.1 of [46], $0 \in \partial F(W, H)$ is equivalent to

$$0 \in \partial_W F(W, H), \text{ and } 0 = \nabla_H F(W, H)$$

namely,

$$0 \in \nabla_W f(W, H) + \partial g_\beta(W) + \partial \iota_+(W), \quad (24a)$$

$$0 = \nabla_H f(W, H). \quad (24b)$$

We call $(W, H)$ a critical point of (11) if it satisfies (24).

In the following, we first present a subsequence convergence result without any assumption. Assuming existence of a full rank limit point, we further show a global sequence convergence result. The proofs of both results involve much technical material and thus are deferred to the appendix for the reader’s convenience.
Theorem 2 (Subsequence convergence). Let \( \{(W^k, H^k)\}_{k=1}^{\infty} \) be the sequence generated from Algorithm ASSL. Any finite limit point of \( \{(W^k, H^k)\}_{k=1}^{\infty} \) is a critical point.

Due to the coercivity of \( g(W) \) and the nonincreasing monotonicity of the objective value, \( \{W^k\} \) must be bounded. However, in general, we cannot guarantee the boundedness of \( \{H^k\} \) because \( XW^k \) may be rank-degenerate (i.e., not full rank). As shown in the next theorem, if we have rank-nondegeneracy of \( XW^k \) in the limit, a stronger convergence result can be established.

Theorem 3 (Global sequence convergence). Let \( \{(W^k, H^k)\}_{k=1}^{\infty} \) be the sequence generated from Algorithm ASSL. If there is a finite limit point \( (\bar{W}, \bar{H}) \) such that \( X\bar{W} \) is full-rank, then the entire sequence \( \{(W^k, H^k)\}_{k=1}^{\infty} \) must converge to \( (\bar{W}, \bar{H}) \).

5. Experimental Studies

In this section, the proposed algorithms ASSL and GLPSL are tested on six benchmark datasets and compared with five state-of-the-art unsupervised feature selection methods.

5.1. Datasets

The six benchmark datasets we use come from different areas, and their characteristics are listed in Table 2. Yale64, WarpPIE, Orl64 and Orlraws\(^2\) are face images, each sample of the datasets representing a face image. Usps\(^3\) is a handwritten digit dataset that contains 9,298 handwritten digit images. Isolet\(^3\) is a speech signal dataset containing 30 speakers’ speech signal of alphabet twice. All datasets are normalized such that the vector corresponding to each feature has unit \( \ell_2 \)-norm.

Table 2: The datasets detail

| Dataset   | \# Instances | \# Features | \# Classes | Type of Data |
|-----------|--------------|-------------|------------|--------------|
| Yale64    | 165          | 4096        | 15         | Face image   |
| WarpPIE   | 210          | 2420        | 10         | Face image   |
| Orl64     | 400          | 4096        | 50         | Face image   |
| Orlraws   | 100          | 10304       | 10         | Face image   |
| Usps      | 9298         | 256         | 10         | Digit image  |
| Isolet    | 1560         | 617         | 26         | Speech signal |

5.2. Experimental Settings

Our algorithms are compared to the following state-of-the-art unsupervised feature selection methods:

1. \textit{LS}: Laplacian score (LS) method uses the Laplacian score to evaluate effectiveness of the features. It selects the features individually that retain the samples’ local similarity specified by a similarity matrix. \cite{7}.

2. \textit{MCFS}: Multi-cluster feature selection (MCFS) is two-step method, and it formulates the feature selection process as a spectral information regression problem with \( \ell_1 \)-norm regularization term \cite{5}.

\(^2\)http://featureselection.asu.edu/datasets.php
\(^3\)http://www.cad.zju.edu.cn/home/dengcai/Data/data.html
3. **UDFS**: Unsupervised discriminative feature selection (UDFS) method combines the data’s local discriminative property and the $\ell_{2,1}$-norm sparse constraint in one convex model to select the features which have the highest power of local discriminative property [13].

4. **RSR**: Regularized self-representation (RSR) feature selection method uses the $\ell_{2,1}$-norm to measure the fitting error and also $\ell_{2,1}$-norm to promote sparsity [25]. Specifically, it solves the following problem:

$$\min_{W} \|X - XW\|_{2,1} + \beta \|W\|_{2,1}. \tag{25}$$

5. **MFFS**: Matrix factorization feature selection (MFFS) method [34] is similar to ours. It performs the subspace learning and feature selection process simultaneously with the orthogonality constraint on the feature selection matrix $W$. Specifically, it solves the following problem:

$$\min_{W,H} \frac{1}{2}\|X - XWH\|_{F}^{2} \text{ s.t. } W \geq 0, H \geq 0, W^\top W = I. \tag{25}$$

There are some parameters we need to set in advance. The number of selected features is taken from $\{20, 30, 40, 50, 60, 70, 80, 90, 100\}$ for all datasets. We use the LPP method in section 3.2.2 to preserve local structure of the data in ASSL because both MCFS and LS use the Laplacian graph, and we set the number of nearest neighbors to $m = 5$ for LS, MCFS, UDFS and ASSL. The parameter $m$ is required by LS, MCFS and ASSL to build a similarity matrix and UDFS to build the local total scatter and between-class scatter matrices. For simplicity, the parameter of local structure preserving term is fixed to be $\mu = 1$ in ASSL for all the tests discussed in Sections 5.3.1 and 5.3.2. We study the sensitivity of ASSL to $\mu$ in Section 5.3.3. The sparsity parameter for UDFS, RSR, and ASSL is tuned from $\{0.01, 0.1, 1, 10, 40, 70, 100\}$. After completing the feature selection process, we use the K-means algorithm [47] to cluster the samples using the selected features. Because the performance of K-means depends on the initial point, we run it 20 times with different random starting points and report the average value.

The compared algorithms are evaluated based on their clustering results. For each sample of all datasets, we set its class number as the cluster number. To measure the clustering performance, we use clustering accuracy (ACC) and normalized mutual information (NMI), which are defined below. Let $p_i$ and $q_i$ be the predicted and true labels of the $i$-th sample, respectively. The ACC is computed as

$$\text{ACC} = \frac{\sum_{i=1}^{n} \delta(q_i, \text{map}(p_i))}{n}, \tag{26}$$

where $\delta(a, b) = 1$ if $a = b$ and $\delta(a, b) = 0$ otherwise, and $\text{map}()$ is a permutation mapping that maps each predicted label to the equivalent true label. We use the Kuhn-Munkres algorithm [48] to realize such a mapping. High value of ACC indicates the predicted labels are close to the true ones, and thus the higher ACC is, the better the clustering results are. The NMI is used to measure the similarity of two clustering results. For two label vectors $P$ and $Q$, it is defined as

$$\text{NMI}(P, Q) = \frac{I(P, Q)}{\sqrt{H(P)H(Q)}}, \tag{27}$$

where $I(P, Q)$ is the mutual information of $P$ and $Q$, $H(P)$ and $H(Q)$ are the entropies of $P$ and $Q$ [49]. In our experiments, $P$ contains the clustering labels using the selected features and $Q$ the true labels of samples in the dataset. Higher value of $\text{NMI}(P, Q)$ implies that $P$ better predicts $Q$. 

14
5.3. **Experimental results**

In this subsection, we report the results of all tested methods. In addition, we study the sensitivity of the parameters present in (11).

### 5.3.1. Performance comparison

In Tables 3 and 4, we present the ACC and NMI values produced by different methods. For each method, we vary the number of selected features among [20, 30, 40, ..., 100] and report the best result. From the tables, we see that ASSL performs the best among all the compared methods except for Yale64, Orl64 and WarpPIE in Table 3 and Yale64, Orl64 in Table 4 for each of which ASSL is the second best. In addition, we see that the greedy method GLPSL performs reasonably well in many cases but can be very bad in some cases such as Usps in both Tables, and this validates our reason to relax (3) and develop ASSL method. Finally, we see that ASSL outperforms MFFS for all datasets, and this is possibly due to the local structure preserving term used in ASSL.

#### Table 3: Clustering results (ACC% ± std%) of different feature selection algorithms on different datasets. The best results are highlighted in bold and the second best results are underlined. (The higher ACC is, the better the result is.)

| Dataset  | Isolet | Yale64 | Orl64 | WarpPIE | Usps | OrIraw |
|----------|--------|--------|-------|---------|------|--------|
| LS       | 55.14 ± 3.15 | 41.25 ± 3.28 | 41.73 ± 1.71 | 32.33 ± 1.03 | 59.79 ± 2.72 | 66.12 ± 6.82 |
| MCFS     | 54.95 ± 3.28 | 44.88 ± 3.72 | 50.75 ± 1.25 | 50.38 ± 2.25 | 66.55 ± 3.11 | 72.43 ± 7.15 |
| UDFS     | 29.60 ± 2.73 | 38.21 ± 3.02 | 40.78 ± 1.03 | 55.57 ± 2.92 | 50.59 ± 1.97 | 65.32 ± 6.18 |
| RSR      | 49.88 ± 3.75 | 45.48 ± 3.34 | 53.24 ± 1.83 | 37.52 ± 2.23 | 62.54 ± 2.34 | 72.54 ± 6.52 |
| MFSS     | 55.39 ± 3.32 | 49.09 ± 3.64 | 50.19 ± 1.64 | 36.57 ± 2.32 | 63.30 ± 3.36 | 73.55 ± 7.68 |
| GLPSL    | 49.05 ± 3.02 | 53.97 ± 3.45 | 41.72 ± 1.05 | 47.52 ± 1.87 | 51.91 ± 2.18 | 72.16 ± 7.03 |
| ASSL     | **62.45 ± 3.58** | **53.45 ± 3.88** | **54.27 ± 1.87** | **52.76 ± 2.12** | **67.24 ± 3.27** | **79.37 ± 7.34** |

#### Table 4: Clustering results (NMI% ± std%) of different feature selection algorithms on different datasets. The best results are highlighted in bold and the second best results are underlined. (The higher NMI is, the better the result is.)

| Dataset  | Isolet | Yale64 | Orl64 | WarpPIE | Usps | OrIraw |
|----------|--------|--------|-------|---------|------|--------|
| LS       | 69.73 ± 1.43 | 46.88 ± 2.07 | 62.61 ± 1.53 | 30.06 ± 2.89 | 56.62 ± 0.95 | 73.38 ± 3.12 |
| MCFS     | 69.82 ± 1.37 | 53.70 ± 1.58 | 69.33 ± 1.62 | 54.37 ± 4.95 | 61.01 ± 0.92 | 83.91 ± 5.53 |
| UDFS     | 44.98 ± 1.02 | 47.40 ± 1.64 | 62.38 ± 1.41 | 54.55 ± 4.38 | 41.31 ± 7.21 | 68.78 ± 3.45 |
| RSR      | 63.47 ± 1.10 | 56.08 ± 1.43 | 72.33 ± 1.75 | 41.81 ± 3.75 | 55.32 ± 1.52 | 83.96 ± 4.35 |
| MFSS     | 72.64 ± 1.73 | 56.17 ± 4.47 | 70.65 ± 1.25 | 40.95 ± 3.39 | 59.11 ± 0.76 | 81.09 ± 4.12 |
| GLPSL    | 65.41 ± 1.23 | **61.39 ± 1.72** | 64.76 ± 1.50 | 53.33 ± 3.89 | 40.98 ± 0.87 | 72.97 ± 3.37 |
| ASSL     | **74.28 ± 1.25** | **58.87 ± 1.65** | **73.02 ± 2.02** | **55.76 ± 4.56** | **61.29 ± 1.25** | **85.65 ± 4.15** |

#### 5.3.2. Compare the performance with all features

To illustrate the effect of feature selection to clustering, we compare the clustering results using all features and selected features given by different methods. Figure 1 plots the ACC value and Figure 2 plots the NMI value with respect to the number of selected features. The baseline corresponds to the results using all features. From the figures, we see that in most cases, the proposed ASSL method gives the best results, and selecting reasonably many features (but far less than the total number of features), it can give comparable and even better clustering results than those by using all features. Hence, the feature selection eliminates the redundancy of the data for clustering purpose. In addition, note that using fewer features can save the clustering time of the K-means method, and thus feature selection can improve both clustering accuracy and efficiency.
Figure 1: The clustering accuracy (ACC) of using all features and selected features by different feature selection algorithms.

Figure 2: The normalized mutual information (NMI) of using all features and selected features by different feature selection algorithms.
5.3.3. On sensitivity of parameters

To further demonstrate the nice performance of the proposed ASSL method, we study its sensitivity with regard to the parameters \(K, \mu\) and \(\beta\) in (11). First, we fix \(\mu = 1\) and vary \(K\) and \(\beta\). Figures 3 and 4 plot the ACC and NMI values given by ASSK for different \(K\) and \(\beta\)'s. From the figures, we see that except for Isolet, ASSL performs stably well for different combinations of \(K\) and \(\beta\), and thus the users can choose the parameters within a large interval to have satisfactory clustering performance. Secondly, we fix \(\beta = 1\) and vary \(K\) and \(\mu\). Figures 5 and 6 plot the ACC and NMI values given by ASSL for different \(K\) and \(\mu\)'s. Again, we see that ASSL performs stably well except for the Isolet dataset.

6. Conclusions

We have proposed a new unsupervised feature selection model, which achieves sparse subspace learning and preserves local structure of the data simultaneously. The model is derived by relaxing an existing combinatorial model with 0-1 variables. A greedy algorithm has been developed, for the first time, to solve the combinatorial problem, and an accelerated block coordinate descent (BCD) method was applied to solve the relaxed continuous task. We have established the global convergence of the BCD method. Extensive numerical tests on real-world data demonstrated that the proposed method outperformed several state-of-the-art unsupervised feature selection methods.
Figure 4: The normalized mutual information (NMI) given by ASSL with different $K$ and $\beta$.

Figure 5: The clustering accuracy (ACC) given by ASSL with different $K$ and $\mu$. 
Appendix A. Proof of Theorem 2

For simplicity, we assume $\omega^k_W = 0$, \forall k, i.e., there is no extrapolation. The case of $\omega^k_W \neq 0$ is more complicated but can be treated similarly with more care taken to handle details; see (41) for example.

The following result is well-known (c.f. Lemma 2.1 of (41))

$$F(W^k, H^k) - F(W^{k+1}, H^k) \geq \frac{L_k}{2} \|W^{k+1} - W^k\|^2_F \geq \frac{L_\mu}{2} \|W^{k+1} - W^k\|^2_F,$$  \hspace{1cm} (A.1)

where

$$L_\mu = \mu \|X^T L X\|_2 > 0.$$ \hspace{1cm} (A.2)

By Lemma 3.1 of (50), we have

$$\frac{1}{2} \|X - XW^{k+1} H^k\|^2_F - \frac{1}{2} \|X - XW^{k+1} H^{k+1}\|^2_F = \frac{1}{2} \|XW^{k+1} H^k - XW^{k+1} H^{k+1}\|^2_F$$  \hspace{1cm} (A.3)

and

$$XW^{k+1} H^k - XW^{k+1} H^{k+1} = U^{k+1} (U^{k+1})^T (XW^{k+1} H^k - X).$$  \hspace{1cm} (A.4)
where $U^{k+1}$ contains the left $r$ leading singular vectors of $XW^{k+1}$ and $r$ is the rank of $XW^{k+1}$.

Note that

$$F(W^{k+1}, H^k) - F(W^{k+1}, H^{k+1}) = \frac{1}{2}\|X - XW^{k+1}H^k\|^2_F - \frac{1}{2}\|X - XW^{k+1}H^{k+1}\|^2_F.$$ 

Hence, summing (A.1) and (A.3) over $k$ and noting nonnegativity of $F$ we obtain

$$\sum_{k=0}^{\infty} \left( \frac{L_w}{2}\|W^{k+1} - W^k\|^2_F + \frac{1}{2}\|XW^{k+1}H^k - XW^{k+1}H^{k+1}\|^2_F \right) \leq F(W^0, H^0),$$

and thus

$$\lim_{k \to \infty} W^{k+1} - W^k = 0. \quad (A.5)$$

and

$$\lim_{k \to \infty} U^{k+1}(U^{k+1})^\top(XW^{k+1}H^k - X) = \lim_{k \to \infty} XW^{k+1}H^k - XW^{k+1}H^{k+1} = 0. \quad (A.6)$$

Combining the two equalities in (A.6), we have

$$\lim_{k \to \infty} U^k(U^k)^\top(XW^kH^k - X) = 0.$$ 

Since $\{XW^k\}$ is bounded and $(XW^k)^\top = (XW^k)^\top U^k(U^k)^\top$, left multiplying $(XW^k)^\top$ in the above equation gives

$$\lim_{k \to \infty} (XW^k)^\top(XW^kH^k - X) = 0. \quad (A.7)$$

Assume $(\hat{W}, \hat{H})$ is a finite limit point of $\{(W^k, H^k)\}_{k=1}^{\infty}$. Then there exists a subsequence $\{W^k, H^k\}_{k \in \mathcal{K}}$ convergent to $(\hat{W}, \hat{H})$. If necessary, taking another subsequence, we can assume $L_w^k \to \hat{L}$ for some $\hat{L} > 0$ as $\mathcal{K} \ni k \to \infty$. From (A.7), it holds that

$$\nabla_H f(\hat{W}, \hat{H}) = (X\hat{W})^\top(X\hat{W}H - X) = 0.$$ 

In addition, from the update rule of $W$, we have

$$W^{k+1} = \arg\min_{W \geq 0} (W, f(W^k, H^k), W - W^k) + \frac{L_w}{2}\|W - W^k\|^2_F + g_\beta(W).$$

Letting $\mathcal{K} \ni k \to \infty$ in the above equation and using (A.5) yield

$$\hat{W} = \arg\min_{W \geq 0} (W, f(\hat{W}, \hat{H}), W - \hat{W}) + \frac{\hat{L}}{2}\|W - \hat{W}\|^2_F + g_\beta(W),$$

which implies

$$0 \in \nabla_W f(\hat{W}, \hat{H}) + \partial g_\beta(W) + \partial \psi(\hat{W}) = \partial F(\hat{W}, \hat{H}).$$

Therefore, $(\hat{W}, \hat{H})$ is a critical point of (11).
Appendix B. Proof of Theorem

For simplicity of notation, we let \( Z^k = (W^k, H^k) \) and \( Z = (\tilde{W}, \tilde{H}) \). In addition, we assume \( \omega_{k}^{W} = 0, \forall k \) as in the proof of Theorem\(^2\). Again, the case of \( \omega_{k}^{W} \neq 0 \) can be shown similarly. Let \( \sigma_{\min}(XW) > 0 \) be the smallest singular value of \( XW \). By the continuity of singular value function and spectral norm of a matrix, there exists \( \delta > 0 \) such that

\[
\sigma_{\min}(XW) \geq \frac{\delta}{2}, \quad \text{and} \quad \|XW\|_2 \leq 2\|XW\|_2, \quad \forall W \in \mathcal{B}(\bar{W}, \delta),
\]

where \( \sigma_{\min}(A) \) denotes the smallest singular value of matrix \( A \), and \( \mathcal{B}(\bar{Z}, \delta) := \{Z : \|Z - \bar{Z}\|_F \leq \delta\} \).

Since \( F \) is a semi-algebraic function, it exhibits the so-called Kurdyka-Łojasiewicz property (c.f. \( \S \)): in a neighborhood \( \mathcal{B}(\bar{Z}, \rho) \), there exists \( \phi(s) = cs^{1-\theta} \) for some \( c > 0 \) and \( 0 \leq \theta < 1 \) such that

\[
\phi'(||F(Z) - F(\bar{Z})||) \text{dist}(0, \partial F(Z)) \geq 1, \quad \text{for any} \quad Z \in \mathcal{B}(\bar{Z}, \rho) \quad \text{and} \quad F(Z) \neq F(\bar{Z}).
\]

Let

\[
F_k = F(Z^k) - F(\bar{Z}), \quad \text{and} \quad \phi_k = \phi(F_k).
\]

Without loss of generality, we assume \( Z^0 \) is sufficiently close to \( \bar{Z} \) such that

\[
2\|Z^0 - \bar{Z}\|_F + 3 \left( \frac{2F_0}{L_\mu} + \frac{8F_0}{\sigma_{\min}^2(XW)} \right) \geq \frac{C_1}{2C_2} \phi_0 < \rho,
\]

where \( L_\mu \) is defined in \( \mathcal{A.2} \), and

\[
C_1 = L_\mu + 2\|\tilde{H}\tilde{H}^\top\|_2\|XX^\top\|_2 + L_\mu, \\
C_2 = \frac{L_\mu}{2} + \frac{\sigma_{\min}^2(XW)}{8}.
\]

In the above equation, \( L_\delta \) is the Lipschitz constant of \( \nabla_W f(W, H) \) in \( \mathcal{B}(\bar{Z}, \delta) \), i.e.,

\[
\|\nabla_W f(\tilde{Z}) - \nabla_W f(\bar{Z})\|_F \leq L_\delta \|\tilde{Z} - \bar{Z}\|_F, \quad \forall \tilde{Z}, \bar{Z} \in \mathcal{B}(\bar{Z}, \delta).
\]

Note that \( L_\delta \) must be finite since \( f(W, H) \) is twice continuous differentiable and \( \mathcal{B}(\bar{Z}, \delta) \) is bounded. Otherwise if \( \mathcal{B.3} \) does not hold, since \( \bar{Z} \) is a limit point of \( \{Z^k\} \), we can take an iterate \( Z^{k_0} \) sufficiently close to \( \bar{Z} \) and let \( Z^{k_0} \) be the new starting point. If necessary, taking a smaller \( \rho \), we assume

\[
\rho + \sqrt{\frac{2F_0}{L_\mu}} \leq \delta,
\]

where \( \delta \) is the quantity used in \( \mathcal{B.1} \).

From \( \mathcal{A.1} \) and \( F_{k+1} \leq F_k \leq \frac{F_1}{2} \), \( \forall k \), we have \( \|W^1 - W^0\|_F \leq \sqrt{\frac{F_1}{L_0}} \) and thus

\[
\|W^1 - \bar{W}\|_F \leq \|W^1 - W^0\|_F + \|W^0 - \bar{W}\|_F \leq \|W^0 - \bar{W}\|_F + \sqrt{\frac{2F_0}{L_\mu}} < \rho \leq \delta.
\]

21
Hence, \( \sigma_{\min}(XW^1) \geq \frac{\sigma_{\min}(X\bar{W})}{2} \) from (B.1a), and
\[
F(W^1, H^0) - F(W^1, H^1) \geq \frac{\sigma_{\min}^2(XW^1)}{2} \|H^1 - H^0\|_F^2 \geq \frac{\sigma_{\min}^2(X\bar{W})}{8} \|H^1 - H^0\|_F^2,
\]
which implies \( \|H^1 - H^0\|_F \leq \frac{8F_0}{\sigma_{\min}^2(X\bar{W})}. \) Therefore,
\[
\|H^1 - \bar{H}\|_F \leq \|H^1 - H^0\|_F + \|H^0 - \bar{H}\|_F \leq \|H^0 - \bar{H}\|_F + \sqrt{\frac{8F_0}{\sigma_{\min}^2(X\bar{W})}}.
\]
Combining (B.8) and (B.9), we have
\[
\|Z^1 - \bar{Z}\|_F \leq \|W^1 - \bar{W}\|_F + \|H^1 - \bar{H}\|_F \leq 2\|Z^0 - \bar{W}\|_F + \sqrt{\frac{2F_0}{L_\mu}} + \sqrt{\frac{8F_0}{\sigma_{\min}^2(X\bar{W})}}.
\]
which together with (B.3) implies \( Z^1 \in \mathcal{B}(\bar{Z}, \rho). \)
Assume that for some integer \( K, Z^k \in \mathcal{B}(\bar{Z}, \rho), \forall 0 \leq k \leq K. \) We go to show \( Z^{k+1} \in \mathcal{B}(\bar{Z}, \rho) \) and thus by induction \( Z^k \in \mathcal{B}(\bar{Z}, \rho), \forall k. \) Note that
\[
0 \in \nabla f(W^k, H^k) + L_W^{-1}(W^k - W^{k-1}) + \partial g_\nu(W^k) + \partial \mu(W^k),
\]
\[
0 = \nabla f(W^k, H^k).
\]
Hence,
\[
\text{dist}(0, \partial F(Z^k)) \leq \|\nabla f(W^k, H^k) - \nabla f(W^{k-1}, H^{k-1})\|_F + L_W^{-1}\|W^k - W^{k-1}\|_F \leq C_1\|Z^k - Z^{k-1}\|_F,
\]
where \( C_1 \) is defined in (B.4). We have
\[
\phi_k - \phi_{k+1} \geq \phi(F_k - F_{k+1}) \quad \text{(from concavity of } \phi) \]
\[
\geq \frac{F_k - F_{k+1}}{C_1\|Z^k - Z^{k-1}\|_F} \quad \text{(from KL property } (B.2) )
\]
\[
\geq \frac{C_2\|Z^{k+1} - Z^k\|_F^2}{C_1\|Z^k - Z^{k-1}\|_F},
\]
where the last inequality follows from (B.5), (A.1) and
\[
F(W^{k+1}, H^k) - F(W^{k+1}, H^{k+1}) \geq \frac{\sigma_{\min}^2(X\bar{W})}{8} \|H^k - H^{k+1}\|_F^2.
\]
Transforming (B.11) gives
\[
C_2\|Z^{k+1} - Z^k\|_F^2 \leq C_1\|Z^k - Z^{k-1}\|_F(\phi_k - \phi_{k+1}) \]
\[
\Rightarrow \sqrt{C_2}\|Z^{k+1} - Z^k\|_F \leq \sqrt{C_1}\|Z^k - Z^{k-1}\|_F(\phi_k - \phi_{k+1})
\]
\[
\Rightarrow \sqrt{C_2}\|Z^{k+1} - Z^k\|_F \leq \frac{\sqrt{C_2}}{2}\|Z^k - Z^{k-1}\|_F \cdot (\phi_k - \phi_{k+1}) + \frac{C_1}{2\sqrt{C_2}}(\phi_k - \phi_{k+1}).
\]

22
Summing the above inequality over $k$ and arranging terms give

$$
\sum_{k=1}^{K} \|Z^{k+1} - Z^k\|_F \leq \|Z^1 - Z^0\|_F + \frac{C_1}{2C_2}(\phi_1 - \phi_{K+1}). \tag{B.12}
$$

Hence,

$$
\|Z^{K+1} - \bar{Z}\|_F \leq \sum_{k=1}^{K} \|Z^{k+1} - Z^k\|_F + \|Z^1 - \bar{Z}\|_F
\leq \|Z^1 - \bar{Z}\|_F + \|Z^1 - Z^0\|_F + \frac{C_1}{2C_2} \phi_0
\leq 2\|Z^1 - Z^0\|_F + \|Z^0 - \bar{Z}\|_F + \frac{C_1}{2C_2} \phi_0
$$

(from (B.8) and (B.9)) \tag{B.13}

which indicates $Z^{K+1} \in B(\bar{Z}, \rho)$. By induction, we have $Z^i \in B(\bar{Z}, \rho)$, $\forall k$, and thus (B.12) holds for all $K$. Therefore, $(Z^i)_{i=1}^{\infty}$ is a Cauchy sequence and converges. Since $\bar{Z}$ is a limit point, it must hold that $\lim_{k \to \infty} Z^k = \bar{Z}$. This completes the proof.

Reference

References

[1] I. Guyon, Feature extraction: foundations and applications, Vol. 207, Springer Science & Business Media, 2006.
[2] L. Wang, H. Cheng, Z. Liu, C. Zhu, A robust elastic net approach for feature learning, Journal of Visual Communication and Image Representation 25 (2) (2014) 313–321.
[3] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, The Journal of Machine Learning Research 3 (2003) 1157–1182.
[4] Q. Gu, Z. Li, J. Han, Joint feature selection and subspace learning, in: IJCAI Proceedings-International Joint Conference on Artificial Intelligence, Vol. 22, 2011, p. 1294.
[5] D. Cai, C. Zhang, X. He, Unsupervised feature selection for multi-cluster data, in: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 2010, pp. 333–342.
[6] R. O. Duda, P. E. Hart, D. G. Stork, Pattern classification, John Wiley & Sons, 2012.
[7] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, in: Advances in neural information processing systems, 2005, pp. 507–514.
[8] X. He, D. Cai, S. Yan, H.-J. Zhang, Neighborhood preserving embedding, in: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on, Vol. 2, IEEE, 2005, pp. 1208–1213.
[9] J. Lu, K. N. Plataniotis, A. N. Venetsanopoulos, Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition, Pattern Recognition Letters 26 (2) (2005) 181–191.
[10] L. Qiao, S. Chen, X. Tan, Sparsity preserving projections with applications to face recognition, Pattern Recognition 43 (1) (2010) 331–341.
[11] Q. Gu, Z. Li, J. Han, Generalized fisher score for feature selection, arXiv preprint arXiv:1202.3725.
[12] J. Lee Rodgers, W. A. Nicewander, Thirteen ways to look at the correlation coefficient, The American Statistician 42 (1) (1988) 59–66.
[13] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, Pattern Analysis and Machine Intelligence, IEEE Transactions on 27 (8) (2005) 1226–1238.
[14] P. Mitra, C. Murthy, S. K. Pal, Unsupervised feature selection using feature similarity, IEEE transactions on pattern analysis and machine intelligence 24 (3) (2002) 301–312.
[15] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, X. Zhou, I2, 1-norm regularized discriminative feature selection for unsupervised learning, in: IJCAI Proceedings-International Joint Conference on Artificial Intelligence, Vol. 22, Citeseer, 2011, p. 1589.
[16] Z. Li, Y. Yang, J. Liu, X. Zhou, H. Lu, Unsupervised feature selection using nonnegative spectral analysis., in: AAAI, 2012.
[17] X. Liu, L. Wang, J. Zhang, J. Yin, H. Liu, Global and local structure preservation for feature selection, Neural Networks and Learning Systems, IEEE Transactions on 25 (6) (2014) 1083–1095.
[18] Z. Zhao, L. Wang, H. Liu, J. Ye, On similarity preserving feature selection, Knowledge and Data Engineering, IEEE Transactions on 25 (3) (2013) 619–632.
[19] H. Cheng, Z. Liu, J. Yang, Sparsity induced similarity measure for label propagation, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 317–324.
[20] H. Cheng, Z. Liu, L. Hou, J. Yang, Sparsity induced similarity and its applications, IEEE Transactions on Circuits and Systems for Video Technology 1.
[21] J. Fang, Y. Shen, H. Li, P. Wang, Pattern-coupled sparse bayesian learning for recovery of block-sparse signals, IEEE Trans. Signal Processing 63 (2) (2015) 360–372.
[22] Z. Zhou, K. Liu, J. Fang, Bayesian compressive sensing using normal product priors.
[23] A. Evgeniou, M. Pontil, Multi-task feature learning, Advances in neural information processing systems 19 (2007) 41.
[24] X.-T. Yuan, X. Liu, S. Yan, Visual classification with multitask joint sparse representation, Image Processing, IEEE Transactions on 21 (10) (2012) 4349–4360.
[25] P. Zhu, W. Zuo, L. Zhang, Q. Hu, S. C. Shiu, Unsupervised feature selection by regularized self-representation, Pattern Recognition 48 (2) (2015) 438–446.
[26] X. Yang, H. T. Shen, F. Nie, R. Ji, X. Zhou, Nonnegative spectral clustering with discriminative regularization., in: AAAI, 2011.
[27] M. Qian, C. Zhai, Robust unsupervised feature selection, in: Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, AAAI Press, 2013, pp. 1621–1627.
[28] X. Jiang, Linear subspace learning-based dimensionality reduction, Signal Processing Magazine, IEEE 28 (2) (2011) 16–26.
[29] L. Bottou, V. Vapnik, Local learning algorithms, Neural computation 4 (6) (1992) 888–900.
[30] D. Cai, X. He, J. Han, Spectral regression for efficient regularized subspace learning, in: Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on, IEEE, 2007, pp. 1–8.
[31] H. Zou, T. Hastie, R. Tibshirani, Sparse principal component analysis, Journal of computational and graphical statistics 15 (2) (2006) 265–286.
[32] B. Moghaddam, Y. Weiss, S. Avidan, Spectral bounds for sparse pca: Exact and greedy algorithms, in: Advances in neural information processing systems, 2005, pp. 915–922.
[33] D. Cai, X. He, J. Han, Spectral regression: A unified approach for sparse subspace learning, in: Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on, IEEE, 2007, pp. 73–82.
[34] S. Wang, W. Pedrycz, Q. Zhu, W. Zhu, Subspace learning for unsupervised feature selection via matrix factorization, Pattern Recognition 48 (1) (2015) 10–19.
[35] S. T. Roweis, L. K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[36] X. Niyogi, Locality preserving projections, in: Neural information processing systems, Vol. 16, MIT, 2004, p. 153.
[37] M. Yuan, Y. Lin, Model selection and estimation in regression with grouped variables, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68 (1) (2006) 49–67.
[38] H. H. Zhang, Y. Liu, Y. Wu, Z. Zhu, Variable selection for the multicategory svm via adaptive sup-norm regularization, Electronic Journal of Statistics 2 (2008) 149–167.
[39] Y. Xu, I. Akrotirianakis, A. Chakraborty, Alternating direction method of multipliers for large-scale multiclass support vector machines, Preprint.
[40] J. A. Tropp, A. C. Gilbert, Signal recovery from random measurements via orthogonal matching pursuit, Information Theory, IEEE Transactions on 53 (12) (2007) 4655–4666.
[41] Y. Xu, W. Yin, A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion, SIAM Journal on imaging sciences 6 (3) (2013) 1758–1789.
[42] A. Beck, M. Teboulle, A fast iterative shrinkage-thresholding algorithm for linear inverse problems, SIAM Journal on Imaging Sciences 2 (1) (2009) 183–202.
[43] Y. Xu, W. Yin, A globally convergent algorithm for nonconvex optimization based on block coordinate update, arXiv preprint arXiv:1410.1386.
[44] N. Parikh, S. Boyd, Proximal algorithms, Foundations and Trends in optimization 1 (3) (2013) 123–231.
[45] R. T. Rockafellar, R. J.-B. Wets, Variational analysis, Vol. 317, Springer Science & Business Media, 2009.
[46] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the kurdyka-lojasiewicz inequality, Mathematics of Operations Research 35 (2) (2010) 438–457.
[47] J. A. Hartigan, M. A. Wong, Algorithm as 136: A k-means clustering algorithm, Applied statistics (1979) 100–108.
[48] L. Lovász, M. Plummer, Matching theory, Vol. 367, American Mathematical Soc., 2009.
[49] R. M. Gray, Entropy and information theory, Springer Science & Business Media, 2011.
[50] Y. Xu, On higher-order singular value decomposition from incomplete data, arXiv preprint arXiv:1411.4324.
[51] J. Bolte, A. Daniilidis, A. Lewis, The lojasiewicz inequality for nonsmooth subanalytic functions with applications to subgradient dynamical systems, SIAM Journal on Optimization 17 (4) (2007) 1205–1223.