Unsupervised Feature Analysis with Class Margin Optimization

Sen Wang[1], Feiping Nie[2], Xiaojun Chang[3(✉)], Lina Yao[4], Xue Li[1], and Quan Z. Sheng[4]

[1] School of ITEE, The University of Queensland, Brisbane, Australia
sen.wang@uq.edu.au, xueli@itee.uq.edu.au
[2] Center for OPTIMAL, Northwestern Polytechnical University, Shaanxi, China
feiping.nie@gmail.com
[3] Center for QCIS, University of Technology Sydney, Ultimo, Australia
cxj273@gmail.com
[4] School of CS, The University of Adelaide, Adelaide, Australia
lina@cs.adelaide.edu.au, michael.sheng@adelaide.edu.au

Abstract. Unsupervised feature selection has been attracting research attention in the communities of machine learning and data mining for decades. In this paper, we propose an unsupervised feature selection method seeking a feature coefficient matrix to select the most distinctive features. Specifically, our proposed algorithm integrates the Maximum Margin Criterion with a sparsity-based model into a joint framework, where the class margin and feature correlation are taken into account at the same time. To maximize the total data separability while preserving minimized within-class scatter simultaneously, we propose to embed K-means into the framework generating pseudo class label information in a scenario of unsupervised feature selection. Meanwhile, a sparsity-based model, $\ell_{2,p}$-norm, is imposed to the regularization term to effectively discover the sparse structures of the feature coefficient matrix. In this way, noisy and irrelevant features are removed by ruling out those features whose corresponding coefficients are zeros. To alleviate the local optimum problem that is caused by random initializations of K-means, a convergence guaranteed algorithm with an updating strategy for the clustering indicator matrix, is proposed to iteratively chase the optimal solution. Performance evaluation is extensively conducted over six benchmark data sets. From our comprehensive experimental results, it is demonstrated that our method has superior performance against all other compared approaches.

Keywords: Unsupervised feature selection · Maximum margin criterion · Sparse structure learning · Embedded K-means clustering

1 Introduction

Over the past few years, data are more than often represented by high-dimensional features in a number of research fields, e.g. data mining, computer

A. Appice et al. (Eds.): ECML PKDD 2015, Part I, LNAI 9284, pp. 383–398, 2015.
DOI: 10.1007/978-3-319-23528-8_24

vision, etc. With the inventions of such many sophisticated data representations, a problem has been never a lack of research attention: How to select the most distinctive features from high-dimensional data for subsequent learning tasks, e.g. classification? To answer this question, we take two points into account. First, the number of selected features should be smaller than the one of all features. Due to a lower dimensional representation, the subsequent learning tasks with no doubt can gain benefit in terms of efficiency [31]. Second, the selected features should have more discriminant power than the original all features. Many previous works have proven that removing those noisy and irrelevant features can improve discriminant power in most cases. In light of advantages of feature selection, different new algorithms have been flourished with various types of applications recently [29,32,33].

According to the types of supervision, feature selection can be generally divided into three categories, i.e. supervised, semi-supervised, and unsupervised feature selection algorithms. Representative supervised feature selection algorithms include Fisher score [6], Relief[11] and its extension, ReliefF [12], information gain [20], etc [25]. Label information of training data points is utilized to guide the supervised feature selection methods to seek distinctive subsets of features with different search strategies, i.e. *complete search, heuristic search, and non-deterministic search*. In the real world, class information is quite limited, resulting in the development of semi-supervised feature selection methods [3,4], in which both labeled and unlabeled data are utilized.

In unsupervised scenarios, feature selection is more challenging, since there is no class information to use for selecting features. In the literature, unsupervised feature selection can be roughly categorized into three groups, i.e. *filter*, *wrapper*, and *embedded methods*. Filter-based unsupervised feature selection methods rank features according to some intrinsic properties of data. Then those features with higher scores are selected for the further learning tasks. The selection is independent to the consequent process. For example, He et al. [8] assume that data from the same class are often close to each other and use the locality preserving power of data, also termed as *Laplacian Score*, to evaluate importance degrees of features. In [30], a unified framework has been proposed for both supervised and unsupervised feature selection schemes using a spectral graph. Tabakhi et al. [23] have proposed an unsupervised feature selection method to select the optimal feature subset in an iterative algorithm, which is based on ant colony optimization. Wrapper-based methods as a more sophisticated way wrap learning algorithms to yield learned results that will be used to select distinctive subsets of features. In [15], for instance, the authors have developed a model that selects relevant features using two backward stepwise selection algorithms without prior knowledges of features. Normally, wrapper-based methods have better performance than filter-based methods, since they use learning algorithms. Unfortunately, the disadvantage is that the computation of wrapper methods is more expensive. Embedded methods are seeking a trade-off between them by integrating feature selection and clustering together into a joint framework. Because clustering algorithms can provide pseudo labels that can reflect

the intrinsic information of data, some works [1,14,26] incorporate different clustering algorithms in objective functions to select features.

Most of the existing unsupervised feature selection methods [8,9,14,19,24,30] rely on a graph, e.g. *graph Laplacian*, to reflect intrinsic relationships among data, labeled and unlabeled. When the number of data is extremely large, the computational burden of constructing a graph Laplacian is significantly heavy. Meanwhile, some traditional feature selection algorithms [6,8] neglect correlations among features. The distinctive features are individually selected according to the importance of each feature rather than taking correlations among features into account. Recently, exploiting feature correlations has attracted much research attention [5,17,18,27,28]. It has proven that discovering feature correlation is beneficial to feature selection.

In this paper, we propose a graph-free method to select features by combining Maximum Margin Criterion with feature correlation mining into a joint framework. Specifically, the method, on one hand, aims to learn a feature coefficient matrix that linearly combines features to maximize the class margins. With the increase of the separability of the entire transformed data by maximizing the total scatter, the proposed method also expects distances between data points within the same class to be minimized after the linear transformation by the coefficient matrix. Since there is no class information can be borrowed from, K-means clustering is jointly embedded in the framework to provide pseudo labels. Inspired by recent feature selection works using sparsity-based model on the regularization term [4], on the other hand, the proposed algorithm learns sparse structural information of the coefficient matrix, with the goal of reducing noisy and irrelevant features by removing those features whose coefficients are zeros. The main contributions of this paper can be summarized as follows:

- The proposed method makes efforts to maximize class margins in a framework, where simultaneously considers the separability of the transformed data and distances between the transformed data within the same class. Besides, a sparsity-based regularization model is jointly applied on the feature coefficient matrix to analyze correlations among features in an iterative algorithm;
- K-means clustering is embedded into the framework generating cluster labels, which can be used as pseudo labels. Both maximizing class margins and learning sparse structures can benefit from generated pseudo labels during each iteration;
- Because the performance of K-means is dominated by the initialization, we propose a strategy to avoid our algorithm rapidly converge to a local optimum, which is largely ignored by most of existing approaches using K-means clustering. Theoretical proof of convergence is also provided.
- We have conducted extensive experiments over six benchmark datasets. The experimental results show that our method has better performance than all the compared unsupervised algorithms.

The rest of this paper is organized as follows: Notations and definitions that are used throughout the entire paper will be given in section 2. Our method will

be elaborated in section 3, followed by proposing its optimization with an algorithm to guarantee the convergence property in section 4. In section 5, extensive experimental results are reported with related analysis. Lastly, the conclusion of this paper will be given in section 6.

2 Notations and Definitions

To give a better understanding of the proposed method, notations and definitions which are used throughout this paper are summarized in this section. Matrices and vectors are written as boldface uppercase letters and boldface lowercase letters, respectively. Given a data set denoted as $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x_n}] \in \mathbb{R}^{d \times n}$, where n is the number of training data and d is the feature dimension. The mean of data is denoted as $\bar{\boldsymbol{x}}$. The feature coefficient matrix, $\boldsymbol{W} \in \mathbb{R}^{d \times d'}$, linearly combines data features as $\boldsymbol{W}^T\boldsymbol{X}$, d' is the feature dimension after the linear transformation. Given a cluster centroid matrix for the transformed data, $\boldsymbol{G} = [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_c] \in \mathbb{R}^{d' \times c}$, its cluster indicator of transformed $\boldsymbol{x}_i$ is represented as $\boldsymbol{u}_i = [u_{i1}, \ldots, u_{ic}]$. c is the number of centroids. If transformed $\boldsymbol{x}_i$ belongs to the j-th cluster, $u_{ij} = 1$, $j = 1, \ldots, c$. Otherwise, $u_{ij} = 0$. Correspondingly, the cluster indicator matrix is $\boldsymbol{U} = [\boldsymbol{u}_1^T, \ldots, \boldsymbol{u}_n^T]^T \in \mathbb{R}^{n \times c}$.

For an arbitrary matrix $\boldsymbol{M} \in \mathbb{R}^{r \times l}$, its $\ell_{2,p}$-norm is defined as:

$$\|\boldsymbol{M}\|_{2,p} = \left[\sum_{i=1}^{r} \left(\sum_{j=1}^{l} M_{ij}^2 \right)^{\frac{p}{2}} \right]^{\frac{1}{p}} \tag{1}$$

The i-th row of $\boldsymbol{M}$ is represented by $\boldsymbol{M}^i$. The between-class, within-class and total scatter matrices of data are respectively defined as:

$$\begin{aligned} \boldsymbol{S_b} &= \sum_{i=1}^{c} n_i (\bar{\boldsymbol{x}}_{\boldsymbol{i}} - \bar{\boldsymbol{x}})(\bar{\boldsymbol{x}}_{\boldsymbol{i}} - \bar{\boldsymbol{x}})^T, \\ \boldsymbol{S_w} &= \sum_{i=1}^{c} \sum_{j=1}^{n_i} (\boldsymbol{x_j} - \bar{\boldsymbol{x}}_i)(\boldsymbol{x_j} - \bar{\boldsymbol{x}}_i)^T, \\ \boldsymbol{S_t} &= \sum_{i=1}^{n} (\boldsymbol{x_i} - \bar{\boldsymbol{x}})(\boldsymbol{x_i} - \bar{\boldsymbol{x}})^T \end{aligned} \tag{2}$$

where n_i is the number of data for the c-th class. $\boldsymbol{S_t} = \boldsymbol{S_w} + \boldsymbol{S_b}$. Other notations and definitions will be explained when they are in use.

3 Proposed Method

We now introduce our proposed method for unsupervised feature selection. To exploit distinctive features, an intuitive way is to find a linear transformation

matrix which can project the data into a new space where the original data are more separable. PCA is the most popular approach to analyze the separability of features. PCA aims to seek directions on which transformed data have max variances. In other words, PCA is to maximize the separability of linearly transformed data by maximizing the covariance: $\max_{\boldsymbol{W}} \sum_{i=1}^{n} (\boldsymbol{W}^T(\boldsymbol{x}_i - \bar{\boldsymbol{x}}))^T(\boldsymbol{W}^T(\boldsymbol{x}_i - \bar{\boldsymbol{x}}))$. Without losing the generality, we assume the data has zero mean, i.e. $\bar{\boldsymbol{x}} = 0$. Recall the definition of total scatter of data, PCA is equivalent to maximize the total scatter of data. However, if only total scatter is considered as a separability measure, the within-class scatter might be also geometrically maximized with the maximization of the total scatter. This is not helpful to distinctive feature discovery. The representative model, LDA, solves this problem by maximizing Fisher criterion: $\max_{\boldsymbol{W}} \frac{\boldsymbol{W}^T \boldsymbol{S_b} \boldsymbol{W}}{\boldsymbol{W}^T \boldsymbol{S_w} \boldsymbol{W}}$. However, LDA and its variants require class information to construct between-class and within-class scatter matrices [2], which is not suitable for unsupervised feature selection. Before we give the objective that can solve the aforementioned problem, we first look at a supervised feature selection framework:

$$\begin{aligned} &\max_{\boldsymbol{W}} \sum_{i=1}^{n} (\boldsymbol{W}^T \boldsymbol{x}_i)^T (\boldsymbol{W}^T \boldsymbol{x}_i) - \alpha \sum_{i=1}^{c} \sum_{j=1}^{n_i} (\boldsymbol{W}^T (\boldsymbol{x_j} - \bar{\boldsymbol{x}}_i))^T (\boldsymbol{W}^T (\boldsymbol{x_j} - \bar{\boldsymbol{x}}_i)) - \beta \Omega(\boldsymbol{W}) \\ &\textbf{s.t.} \quad \boldsymbol{W}^T \boldsymbol{W} = \boldsymbol{I}, \end{aligned} \tag{3}$$

where α and β are regularization parameters. In this framework, the first term is to maximize the total scatter, while the second term is to minimize the within-class scatter. The third part is a sparsity-based regularization term which controls the sparsity of $\boldsymbol{W}$. This model is quite similar with the classical LDA-based methods. Due to there is no class information in the unsupervised scenario, we need virtual labels to minimize the distances between data within the same class while maximize the total separability at the same time. To achieve this goal, we apply K-means clustering in our framework to replace the ground truth by generating cluster indicators of data. Given c centroids $\boldsymbol{G} = [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_c] \in \mathbb{R}^{d' \times c}$, the objective function of the traditional K-means algorithm aims to minimize the following function:

$$\begin{aligned} &\sum_{i=1}^{c} \sum_{\boldsymbol{y}_j \in \mathcal{Y}_i} (\boldsymbol{y}_j - \boldsymbol{g}_i)^T (\boldsymbol{y}_j - \boldsymbol{g}_i) \\ =&\sum_{i=1}^{n} (\boldsymbol{y}_i - \boldsymbol{G}\boldsymbol{u}_i^T)^T (\boldsymbol{y}_i - \boldsymbol{G}\boldsymbol{u}_i^T), \end{aligned} \tag{4}$$

where $\boldsymbol{y}_i = \boldsymbol{W}^T \boldsymbol{x}_i$. Note that K-means is used to assign cluster labels, which are used as pseudo labels, to minimize the within-class scatter after the linear transformation by $\boldsymbol{W}$. Then, we can substitute (4) into (3):

$$\max_{\boldsymbol{W}} \sum_{i=1}^{n} (\boldsymbol{W}^T \boldsymbol{x}_i)^T (\boldsymbol{W}^T \boldsymbol{x}_i) - \alpha \sum_{i=1}^{n} (\boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{G}\boldsymbol{u}_i^T)^T (\boldsymbol{W}^T \boldsymbol{x}_i - \boldsymbol{G}\boldsymbol{u}_i^T) - \beta\Omega(\boldsymbol{W})$$
$$\textbf{s.t.} \quad \boldsymbol{W}^T \boldsymbol{W} = \boldsymbol{I}, \tag{5}$$

As mentioned above, the sparsity-based regularization term has been widely used to find out correlated structures among features. The motivation behind this is to exploit sparse structures of the feature coefficient matrix. By imposing the sparse constraint, some of the rows of the feature coefficient matrix shrink to zeros. Those features corresponding to non-zero coefficients are selected as the distinctive subset of features. In this way, noisy and redundant features can be removed. This sparsity-based regularization has been applied in various problems. Inspired by the *"shrinking to zero"* idea, we utilize a sparsity model to uncover the common structures shared by features. To achieve that goal, we propose to minimize the $\ell_{2,p}$-norm of the coefficient matrix, $\|\boldsymbol{W}\|_{2,p}, (0 < p < 2)$. From the definition of $\|\boldsymbol{W}\|_{2,p}$ in (1), outliers or negative impact of the irrelevant $\boldsymbol{w^i}$'s are suppressed by minimizing the $\ell_{2,p}$-norm. Note that p is a parameter that controls the degree of correlated structures among features. The lower p is, the more shared structures among are expected to exploit. After a number of optimization steps, the optimal feature coefficient matrix, $\boldsymbol{W}$, can be obtained. Thus, we impose the $\ell_{2,p}$-norm on the regularization term and re-write the objective function in a matrix representation as follows:

$$\max_{\boldsymbol{W},\boldsymbol{G},\boldsymbol{U}} Tr(\boldsymbol{W}^T \boldsymbol{S_t} \boldsymbol{W}) - \alpha \|\boldsymbol{W}^T \boldsymbol{X} - \boldsymbol{G}\boldsymbol{U}^{\boldsymbol{T}}\|_F^2 - \beta \|\boldsymbol{W}\|_{2,p}$$
$$\textbf{s.t.} \quad \boldsymbol{W}^T \boldsymbol{W} = \boldsymbol{I}, \tag{6}$$

where $\boldsymbol{U}$ is an indicator matrix. $Tr(\cdot)$ is trace operator, while $\|\cdot\|_F^2$ is the Frobenius norm of a matrix. Our proposed method integrates the Maximum Margin Criterion and sparse regularization into a joint framework. Embedding K-means into the framework not only minimizes the distances between within-class data while maximizing total data separability, but also provides cluster labels. The cluster centroids generated by K-means can further guide the sparse structure learning on the feature coefficient matrix in each iterative step of our solution, which will be explained in the next section. We name this method for the unsupervised feature analysis with class margin optimization as **UFCM**.

4 Optimization

In this section, we present our solution to the objective function in (6). Since the $\ell_{2,p}$-norm is used to exploit sparse structures, the objective function cannot be solved in a closed form. Meanwhile, the objective function is not jointly convex with respect to three variables, i.e. $\boldsymbol{W}, \boldsymbol{G}, \boldsymbol{U}$. Thus, we propose to solve the problem as follows.

We define a diagonal matrix $\boldsymbol{D}$ whose diagonal entries are defined as:

$$\boldsymbol{D}^{ii} = \frac{1}{\frac{2}{p}\|\boldsymbol{w}^i\|_2^{2-p}}. \tag{7}$$

The objective function in (6) is equivalent to:

$$\begin{aligned} &\max_{\boldsymbol{W},\boldsymbol{G},\boldsymbol{U}} Tr(\boldsymbol{W}^T\boldsymbol{S_t}\boldsymbol{W}) - \alpha\|\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{G}\boldsymbol{U}^{\boldsymbol{T}}\|_F^2 - \beta Tr(\boldsymbol{W}^T\boldsymbol{D}\boldsymbol{W}) \\ &\quad \textbf{s.t.} \quad \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I} \end{aligned} \tag{8}$$

We propose to optimize the objective function in two steps in each iteration as follows:

(1) Fix $\boldsymbol{W}, \boldsymbol{G}$ and optimize $\boldsymbol{U}$:

When $\boldsymbol{W}$ is fixed, the first and third terms can be viewed as constants. While the second term can be viewed as the objective function of K-means, assigning cluster labels to each data. Also, the cluster centroid matrix $\boldsymbol{G} = [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_c]$ is also fixed, the optimal $\boldsymbol{U}$ is:

$$U_{ij} = \begin{cases} 1, & j = \underset{k}{\operatorname{argmin}} \|\boldsymbol{W}^T\boldsymbol{x}_i - \boldsymbol{g}_k\|_F^2, \\ 0, & \text{Otherwise.} \end{cases} \tag{9}$$

This is equivalent to perform K-means on the transformed data, $\boldsymbol{W}^T\boldsymbol{X}$, which means the solution is unique.

(2) Fix $\boldsymbol{U}$ and optimize $\boldsymbol{W}, \boldsymbol{G}$:

After fixing the indicator matrix, $\boldsymbol{U}$, we set the derivative of Equation (8) with respect to $\boldsymbol{G}$ equal to 0:

$$\begin{aligned} -\alpha\frac{\partial Tr(\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{G}\boldsymbol{U}^T)^T(\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{G}\boldsymbol{U}^T)}{\partial \boldsymbol{G}} &= 0 \\ \Rightarrow -2\alpha\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{U} + 2\alpha\boldsymbol{G}\boldsymbol{U}^T\boldsymbol{U} &= 0 \\ \Rightarrow \boldsymbol{G} = \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1} & \end{aligned} \tag{10}$$

Substituting Equation (10) into Equation (8), we have:

$$\begin{aligned} &Tr(\boldsymbol{W}^T\boldsymbol{S_t}\boldsymbol{W}) - \alpha\|\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^{\boldsymbol{T}}\|_F^2 - \beta Tr(\boldsymbol{W}^T\boldsymbol{D}\boldsymbol{W}) \\ &= \alpha Tr\left((\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^{\boldsymbol{T}} - \boldsymbol{W}^T\boldsymbol{X})(\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^{\boldsymbol{T}})^T\right) \\ &\quad + Tr(\boldsymbol{W}^T\boldsymbol{S_t}\boldsymbol{W}) - \beta Tr(\boldsymbol{W}^T\boldsymbol{D}\boldsymbol{W}) \\ &= \alpha Tr\left(\boldsymbol{W}^T\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^T\boldsymbol{X}^T\boldsymbol{W} - \boldsymbol{W}^T\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{W}\right) \\ &\quad + Tr(\boldsymbol{W}^T\boldsymbol{S_t}\boldsymbol{W}) - \beta Tr(\boldsymbol{W}^T\boldsymbol{D}\boldsymbol{W}) \\ &= Tr[\boldsymbol{W}^T(\boldsymbol{S_t} + \alpha\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^T\boldsymbol{X}^T - \alpha\boldsymbol{X}\boldsymbol{X}^T - \beta D)\boldsymbol{W}] \end{aligned} \tag{11}$$

Thus, the objective function becomes:

$$\begin{aligned} &\max_{\boldsymbol{W}} Tr[\boldsymbol{W}^T(\boldsymbol{S_t} + \alpha\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^T\boldsymbol{X}^T - \alpha\boldsymbol{X}\boldsymbol{X}^T - \beta D)\boldsymbol{W}] \\ &\quad \textbf{s.t.} \quad \boldsymbol{W}^T\boldsymbol{W} = \boldsymbol{I} \end{aligned} \tag{12}$$

The objective function can be then solved by performing eigen-decomposition of the following formula:

$$\boldsymbol{S_t} + \alpha\boldsymbol{X}\boldsymbol{U}(\boldsymbol{U}^T\boldsymbol{U})^{-1}\boldsymbol{U}^T\boldsymbol{X}^T - \alpha\boldsymbol{X}\boldsymbol{X}^T - \beta D \tag{13}$$

Algorithm 1. Unsupervised Feature Analysis with Class Margin Optimization.

Input: Data matrix $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d \times n}$ and parameters α and β.
Output: Feature coefficient matrix $\boldsymbol{W}$ and cluster indicator matrix $\boldsymbol{U}$.
1: Initialize $\boldsymbol{W}$ by PCA on $\boldsymbol{X}$;
2: Initialize $\boldsymbol{U}$ by K-means on $\boldsymbol{W}^T\boldsymbol{X}$;
3: **repeat**
4: Compute $\boldsymbol{D}$ according to (7);
5: Update $\boldsymbol{U}$ according to (14);
6: Update $\boldsymbol{W}$ by eigen-decomposition of (13);
7: Update $\boldsymbol{G}$ according to (10);
8: **until** Convergence

The optimal $\boldsymbol{W}$ can be determined by choosing d' eigenvectors corresponding to d' largest eigenvalues, $d' \leq d$. Our proposed method can be solved by above steps in an iterative algorithm. Each step can obtain the corresponding optimum. As the cluster indicator matrix $\boldsymbol{U}$ is initialized by K-means, the performance of our algorithm is determined by the initialization of K-means. To alleviate the local optimum problem, an update strategy for $\boldsymbol{U}$ is demanded. Generally speaking, we randomly initialize $\boldsymbol{U}$ a number of times and make comparisons according to the second term in Equation (6). Then we choose how to update the indicator matrix. Specifically, the optimal $\boldsymbol{U}_i^*$ and $\boldsymbol{W}_i^*$ has been derived in the i-th iteration. In the $(i+1)$-th iteration, we first randomly initialize $\boldsymbol{U}$ r times ($r = 10$ in our experiment) and combine the derived $\boldsymbol{U}_i^*$ in the i-th iteration as an updating candidate set: $\tilde{\boldsymbol{U}}_{i+1} = [\boldsymbol{U}_{i+1}^0, \boldsymbol{U}_{i+1}^1, \ldots, \boldsymbol{U}_{i+1}^r]$, $\boldsymbol{U}_{i+1}^0 = \boldsymbol{U}_i^*$. According to $\|\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{G}U^T\|_F^2$, the candidate, which yields the smallest value, is chosen to update $\boldsymbol{U}_{i+1}^*$:

$$\boldsymbol{U}_{i+1}^* = \tilde{\boldsymbol{U}}_{i+1}^j, \qquad j = \operatorname*{argmin}_j \|\boldsymbol{W}^T\boldsymbol{X} - \boldsymbol{G}(\tilde{\boldsymbol{U}}_{i+1}^j)^T\|_F^2 \tag{14}$$

where j is the index of candidate set, $j = 0, 1, \ldots, r$. In this way, we compare the derived cluster indicator matrix with r randomly initialized counterparts to alleviate the local optimum problem. We summarize the solution in Algorithm 1 which outputs the learned feature coefficient matrix $\boldsymbol{W}$ to select distinctive features.

From Algorithm 1, it can be seen that the most computational operation is the eigen-decomposition in Equation (13). The computational complexity is $O(d^3)$. If the dimensionality of the data, d, is very high, dimensionality reduction is desirable. To analyze the convergence of our proposed method, the following proposition and its proof are given.

Proposition 1. *Algorithm 1 monotonically increases the objective function in Equation (6) until convergence.*

Proof. Assuming that, in the i-th iteration, the transformation matrix $\boldsymbol{W}$ and cluster centroid matrix $\boldsymbol{G}$ have been derived as $\boldsymbol{W}_i$ and $\boldsymbol{G}_i$. In the $(i+1)$-th

iteration step, we use $\boldsymbol{W}_i$ and $\boldsymbol{G}_i$ to update $\boldsymbol{U}_{i+1}$ according to the updating strategy in (14). We can have the following inequality:

$$\begin{aligned} &Tr(\boldsymbol{W}_i^T \boldsymbol{S_t} \boldsymbol{W}_i) - \alpha \|\boldsymbol{W}_i^T \boldsymbol{X} - \boldsymbol{G}_i \boldsymbol{U}_i^T\|_F^2 - \beta \|\boldsymbol{W}_i\|_{2,p} \\ \leq &Tr(\boldsymbol{W}_i^T \boldsymbol{S_t} \boldsymbol{W}_i) - \alpha \|\boldsymbol{W}_i^T \boldsymbol{X} - \boldsymbol{G}_i \boldsymbol{U}_{i+1}^T\|_F^2 - \beta \|\boldsymbol{W}_i\|_{2,p} \end{aligned} \tag{15}$$

Similarly, when $\boldsymbol{U}_{i+1}$ is fixed to optimize $\boldsymbol{W}$ and $\boldsymbol{G}$ in the $(i+1)$-th iteration, the following inequality can be obtained according to Equation (12):

$$\begin{aligned} &Tr(\boldsymbol{W}_i^T \boldsymbol{S_t} \boldsymbol{W}_i) - \alpha \|\boldsymbol{W}_i^T \boldsymbol{X} - \boldsymbol{G}_i \boldsymbol{U}_{i+1}^T\|_F^2 - \beta \|\boldsymbol{W}_i\|_{2,p} \\ \leq &Tr(\boldsymbol{W}_{i+1}^T \boldsymbol{S_t} \boldsymbol{W}_{i+1}) - \alpha \|\boldsymbol{W}_{i+1}^T \boldsymbol{X} - \boldsymbol{G}_{i+1} \boldsymbol{U}_{i+1}^T\|_F^2 - \beta \|\boldsymbol{W}_{i+1}\|_{2,p} \end{aligned} \tag{16}$$

After combining Equation (15) and (16) together, it indicates that the proposed algorithm will monotonically increase the objective function in each iteration. It is worth noting that the algorithm is alleviating the local optimum problem raised by random initializations of K-means, rather than completely solving it. However, our algorithm can avoid to rapidly converge to a local optimum and may converge to the global optimal solution.

5 Experiments

In this section, experimental results will be presented together with related analysis. We compare our method with seven approaches over six benchmark datasets. Besides, we also conduct experiments to evaluate performance variations in different aspects. They are including the impact of different selected feature numbers, the validation of feature correlation analysis, and parameter sensitivity analysis. Lastly, the convergence demonstration is shown.

5.1 Experiment Setup

In the experiments, we have compared our method with seven approaches as follows:

- **All Features:** All original variables are preserved as the baseline in the experiments.
- **Max Variance:** Features are ranked according to the variance magnitude of each feature in a descending order. The highest ranked features are selected.
- **Spectral Feature Selection (SPEC)** [30]**:** This method employs a unified framework to select features one by one based on spectral graph theory.
- **Multi-Cluster Feature Selection (MCFS)** [1]**:** This unsupervised approach selects those features who make the multi-cluster structure of the data preserved best. Features are selected using spectral regression with the ℓ_1-norm regularization.
- **Robust Unsupervised Feature Selection (RUFS)** [19]**:** RUFS jointly performs robust label learning and robust feature learning. To achieve this, robust orthogonal nonnegative matrix factorization is applied to learn labels while the $\ell_{2,1}$-norm minimization is simultaneously utilized to learn the features.

- **Nonnegative Discriminative Feature Selection (NDFS)** [14]**:** NDFS exploits local discriminative information and feature correlations simultaneously. Besides, the manifold structure information is also considered jointly.
- **Laplacian Score (LapScore)** [8]**:** This method learns and selects distinctive features by evaluating their powers of locality preserving, which is also called Laplacian Score.

All the parameters (if any) are tuned in the range of $\{10^{-3}, 10^{-1}, 10^{1}, 10^{3}\}$ for each algorithm mentioned above and the best results are reported. The size of the neighborhood is set to 5 for any algorithm based on spectral clustering. The number of random initializations required in the update strategy in (14), is set at 10 in the experiment. To measure the performance, two metrics have been used: *Clustering Accuracy (ACC)* and *Normalized Mutual Information (NMI)*.

For a data point x_i, its ground truth label is denoted as p_i and its clustering label that is produced from a clustering algorithm, is represented as q_i. Then, ACC metric over a data set with n data points is defined as follows:

$$ACC = \frac{\sum_{i=1}^{n} \delta(p_i, map(q_i))}{n}, \tag{17}$$

where $\delta(x, y) = 1$ if $x = y$ and $\delta(x, y) = 0$ otherwise. $map(x)$ is the *best mapping function* which permutes clustering labels to match the ground truth labels using the Kuhn-Munkres algorithm. A larger ACC means better performance.

According to the definition in [22], NMI is defined as:

$$NMI = \frac{\sum_{l=1}^{c} \sum_{h=1}^{c} t_{l,h} log(\frac{n \times t_{l,h}}{t_l \tilde{t_h}})}{\sqrt{(\sum_{l=1}^{c} t_l log \frac{t_l}{n})(\sum_{h=1}^{c} \tilde{t_h} log \frac{\tilde{t_h}}{n})}}, \tag{18}$$

where t_l is the number of data points in the l-th cluster, $1 \leq l \leq c$, which is generated by a clustering algorithm. While $\tilde{t_h}$ denotes the number of data points in the h-th ground truth cluster. $t_{l,h}$ is the number of data points which are in the intersection of the l-th and h-th clusters. Similarly, a larger NMI means better performance.

The performance evaluations are performed over six benchmark datasets as follows:

- **COIL20** [16]**:** It contains 1,440 gray-scale images of 20 objects (72 images per object) under various poses. The objects are rotated through 360 degrees and taken at the interval of 5 degrees.
- **MNIST** [13]**:** It is a large-scale dataset of handwritten digits, which has been widely used as a test bed in data mining. The dataset contains 60,000 training images and 10,000 testing images. In this paper, we use its subclass version, MNIST-S, in which one handwritten digit image per ten images, for each class, is randomly sampled from the MNIST database. There are 6,996 handwritten images with a resolution of 28×28.
- **ORL** [21]**:** This data set which is used as a benchmark for face recognition, consists of 40 different subjects with 10 images each. We also resize each image to 32 × 32.

Table 1. Summary of data sets.

	COIL20	MNIST	ORL	UMIST	USPS	YaleB
Number of data	1,440	6,996	400	564	9,298	2,414
Number of classes	20	10	40	20	10	38
Feature dimensions	1,024	784	1,024	644	256	1,024

- **UMIST:** UMIST, which is also known as the Sheffield Face Database, consists of 564 images of 20 individuals. Each individual is shown in a variety of poses from profile to frontal views.
- **USPS** [10]**:** This dataset collects 9,298 images of handwritten digits (0-9) from envelops by the U.S. Postal Service. All images have been normalized to the same size of 16 × 16 pixels in gray scale.
- **YaleB** [7]**:** It consists of 2,414 frontal face images of 38 subjects. Different lighting conditions have been considered in this dataset. All images are reshaped into 32 × 32 pixels.

The pixel value in data is used as the feature. Details of data sets that are used in this paper are summarized in Table 1.

5.2 Experimental Results

To compare the performance of our proposed algorithm with others, we repeatedly perform the test five times and report the average performance results (ACC and NMI) with standard deviations in Tables 2 and 3. It is observed that our proposed method consistently achieves better performance than all other compared approaches across all the data sets. Besides, it is worth noting that our method is superior to those state-of-the-art counterparts that rely on a graph Laplacian (SPEC, RUFS, NDFS, LapScore).

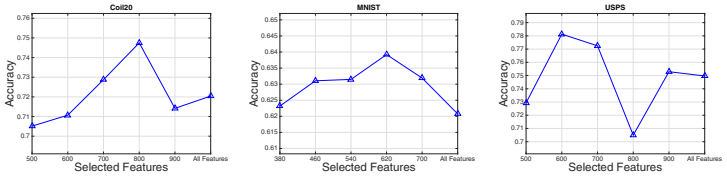
We study how the number of selected features can affect the performance by conducting an experiment whose results are shown in Figure 1. From the figure, performance variations with respect to the number of selected features using the proposed algorithm over three data sets, including COIL20, MNIST, and USPS, have been illustrated. We only adopt ACC as the metric. Some observations can be obtained: 1) When the number of selected features is small, e.g. 500 on each data set, the accuracy value is relatively small. 2) With the increase of selected features, performance can peak at a certain point. For example, the performance of our algorithm peaks at 0.7475 on COIL20 when the number of selected features increases to 800. Similarly, 0.6392 (800 selected features) and

Table 2. Performance comparison ($ACC \pm STD$).

	COIL20	MNIST	ORL	UMIST	USPS	YaleB
AllFea	0.7051 ± 0.0294	0.6009 ± 0.0063	0.6675 ± 0.0112	0.4800 ± 0.0115	0.7139 ± 0.0272	0.1261 ± 0.0025
MaxVar	0.7124 ± 0.0191	0.6239 ± 0.0100	0.6965 ± 0.0121	0.4984 ± 0.0141	0.7165 ± 0.0186	0.1291 ± 0.0042
SPEC	0.7105 ± 0.0116	0.6254 ± 0.0024	0.6645 ± 0.0065	0.4824 ± 0.0077	0.7037 ± 0.0315	0.1307 ± 0.0049
MCFS	0.7355 ± 0.0050	0.6299 ± 0.0037	0.7055 ± 0.0048	0.5239 ± 0.0038	0.7634 ± 0.0138	0.1355 ± 0.0043
RUFS	0.7365 ± 0.0024	0.6294 ± 0.0028	0.6920 ± 0.0033	0.5110 ± 0.0091	0.7659 ± 0.0076	0.1795 ± 0.0032
NDFS	0.7368 ± 0.0074	0.6291 ± 0.0016	0.7050 ± 0.0031	0.5243 ± 0.0028	0.7630 ± 0.0124	0.1315 ± 0.0034
LapScore	0.7126 ± 0.0249	0.6214 ± 0.0054	0.7100 ± 0.0117	0.5092 ± 0.0062	0.7089 ± 0.0324	0.1255 ± 0.0025
Ours	**0.7475±0.0076**	**0.6392 ± 0.0056**	**0.7210 ± 0.0052**	**0.5343 ± 0.0062**	**0.7813 ± 0.007**	**0.1886 ± 0.0043**

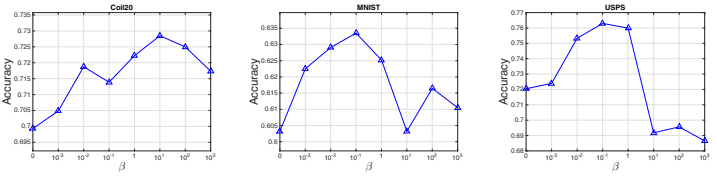
Table 3. Performance comparison ($NMI \pm STD$).

	COIL20	MNIST	ORL	UMIST	USPS	YaleB
AllFea	0.7884 ± 0.0157	0.5162 ± 0.0027	0.8265 ± 0.0129	0.6715 ± 0.0069	0.6305 ± 0.0029	0.1968 ± 0.0017
MaxVar	0.7932 ± 0.0071	0.5314 ± 0.0063	0.8424 ± 0.0085	0.6825 ± 0.0063	0.6361 ± 0.0021	0.2123 ± 0.0040
SPEC	0.7866 ± 0.0061	0.5367 ± 0.0035	0.8232 ± 0.0021	0.6753 ± 0.0114	0.6215 ± 0.0073	0.2071 ± 0.0027
MCFS	0.8066 ± 0.0025	0.5367 ± 0.0003	0.8460 ± 0.0025	0.7005 ± 0.0053	0.6419 ± 0.0015	0.2024 ± 0.0033
RUFS	0.8045 ± 0.0025	0.5374 ± 0.0021	0.8430 ± 0.0044	0.6898 ± 0.0035	0.6468 ± 0.0027	0.2845 ± 0.0040
NDFS	0.8062 ± 0.0058	0.5376 ± 0.0004	0.8458 ± 0.0026	0.6981 ± 0.0054	0.6452 ± 0.0054	0.2048 ± 0.0041
LapScore	0.7920 ± 0.0101	0.5308 ± 0.0065	0.8421 ± 0.0006	0.6924 ± 0.0027	0.6291 ± 0.0047	0.1945 ± 0.0018
Ours	**0.8119 ± 0.0035**	**0.5422 ± 0.0018**	**0.8518 ± 0.0027**	**0.7112 ± 0.0033**	**0.6535 ± 0.0022**	**0.2959 ± 0.0043**



(a) COIL20 (b) MNIST (c) USPS

Fig. 1. Performance variation results with respect to the number of selected features using the proposed algorithm over three data sets, COIL20, MNIST, and USPS.



(a) COIL20 (b) MNIST (c) USPS

Fig. 2. Performance variation results with respect to different values of regularization parameter, βs, over three data sets, COIL20, MNIST, and USPS.

0.7813 (600 selected features) are observed on MNIST and USPS, respectively. 3) When all features are in use, the performance is worse than the best. Similar trends can be also observed on the other data sets. It is concluded that our algorithm can select distinctive features.

To demonstrate exploiting feature correlation is beneficial to the performance, we conduct an experiment in which parameters α and p are both fixed at 1. β varies in a range of $[0, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}]$. The performance variation results with respect to different βs are plotted in Figure 2. The experiment is conducted over three data sets, i.e. COIL20, MNIST, and USPS. From the results, we can observe that the performance is relatively low, when there is no correlation exploiting in the framework, i.e. $\beta = 0$. The performance always peaks at a certain point when a proper degree of sparsity is imposed to the regularization term. For example, the performance is only 0.6993 when $\beta = 0$ on COIL20. The performance increases to 0.7285 when $\beta = 10^{1}$. Similar observations are also obtained on the other data sets. We can conclude that sparse structure learning on feature coefficient matrix contributes to the performance of our unsupervised feature selection method.

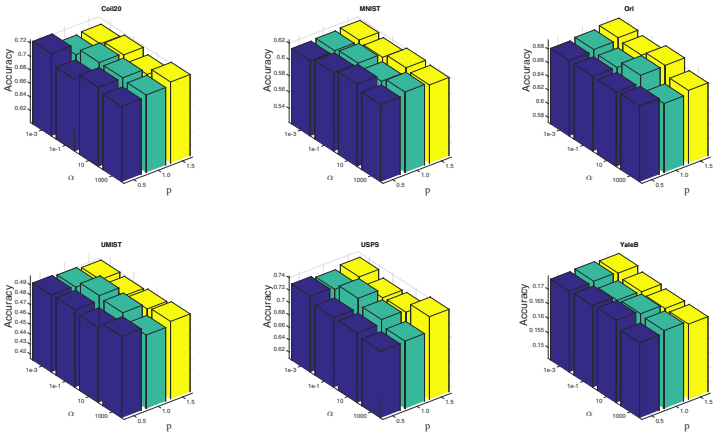


Fig. 3. Performance variation results under different combinations of αs and ps. β is fixed at 10^{-1}.

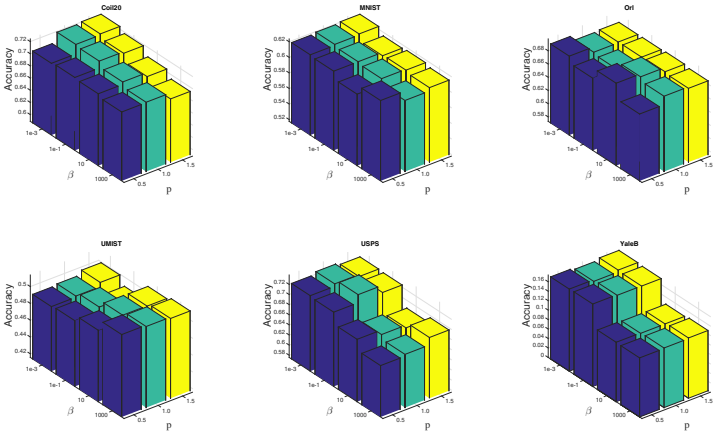
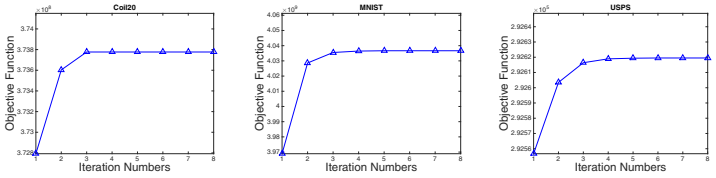


Fig. 4. Performance variation results under different combinations of βs and ps. α is fixed at 10^{-1}.

5.3 Studies on Parameter Sensitivity and Convergence

There are three parameters in our algorithms, which are denoted as α, β and p in (6). α and β are two regularization parameters while p controls the degree of sparsity. To investigate the sensitivity of the parameters, we conduct an experiment to study



(a) COIL20 (b) MNIST (c) USPS

Fig. 5. Objective function values of our proposed objective function in (6) over three data sets, COIL20, MNIST, and USPS.

how they exert influences on performance. Firstly, we fix $\beta = 10^{-1}$ and derive the performance variations under different combinations of αs and ps in Figure 3. Secondly, α is fixed at 10^{-1}. The performance variation results with respect to different βs and ps are shown in Figure 4. Both α and β vary in a range of $[10^{-3}, 10^{-1}, 10^{1}, 10^{1}]$. While p changes in $[0.5, 1.0, 1.5]$. We only take ACC as the metric.

To validate that our algorithm will monotonically increase the objective function value in (6), we conduct an experiment to demonstrate this fact. In this experiment, all parameters (α, β, and p) in (6) are fixed at 1. The objective function values and corresponding iteration numbers are drawn in Figure 5. We take COIL20, MNIST, and USPS as examples. Similar observations can be also obtained on the other data sets. From the figure, it can be seen that our algorithm converges to the optimum, usually within eight iteration steps, over three data sets. We can then conclude that the proposed method is efficient and effective.

6 Conclusion

In this paper, an unsupervised feature selection approach has been proposed by using the Maximum Margin Criterion and the sparsity-based model. More specifically, the proposed method seeks to maximize the total scatter on one hand. On the other hand, the within-class scatter is simultaneously considered to minimize. Since there is no label information in an unsupervised scenario, K-means clustering is embedded into the framework jointly. Advantages can be summarized as twofold: First, pseudo labels generated by K-means clustering is beneficial to maximizing class margins in each iteration step. Second, pseudo labels can guide the sparsity-based model to exploit sparse structures of the feature coefficient matrix. Noisy and uncorrelated features can be therefore removed. Since the objective function is non-convex for all variables, we have proposed an algorithm with a guaranteed convergence property. To avoid to rapidly converge to a local optimum which is caused by K-means, we applied an updating strategy to alleviate the problem. In this way, our proposed method might converge to the global optimum. Extensive experimental results have shown that our method has superior performance against all other compared approaches over six benchmark data sets.

Acknowledgments. This work was supported by Australian Research Council Discovery Project (DP140100104). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Australian Research Council.

References

1. Cai, D., Zhang, C., He, X.: Unsupervised feature selection for multi-cluster data. In: SIGKDD (2010)
2. Chang, X., Nie, F., Wang, S., Yang, Y.: Compound rank-k projections for bilinear analysis. IEEE Trans. Neural Netw. Learning Syst. (2015)
3. Chang, X., Nie, F., Yang, Y., Huang, H.: A convex formulation for semi-supervised multi-label feature selection. In: AAAI (2014)
4. Chang, X., Shen, H., Wang, S., Liu, J., Li, X.: Semi-supervised feature analysis for multimedia annotation by mining label correlation. In: Tseng, V.S., Ho, T.B., Zhou, Z.-H., Chen, A.L.P., Kao, H.-Y. (eds.) PAKDD 2014, Part II. LNCS, vol. 8444, pp. 74–85. Springer, Heidelberg (2014)
5. Chang, X., Yang, Y., Hauptmann, A.G., Xing, E.P., Yu, Y.: Semantic concept discovery for large-scale zero-shot event detection. In: IJCAI (2015)
6. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification. John Wiley & Sons (2012)
7. Georghiades, A.S., Belhumeur, P.N., Kriegman, D.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **23**(6), 643–660 (2001)
8. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. In: NIPS (2005)
9. Hou, C., Nie, F., Li, X., Yi, D., Wu, Y.: Joint embedding learning and sparse regression: A framework for unsupervised feature selection. IEEE T. Cybernetics **44**(6), 793–804 (2014)
10. Hull, J.J.: A database for handwritten text recognition research. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **16**(5), 550–554 (1994)
11. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: IWML (1992)
12. Kononenko, I.: Estimating attributes: analysis and extensions of relief. In: Bergadano, F., De Raedt, L. (eds.) ECML 1994. LNCS, vol. 784, pp. 171–182. Springer, Heidelberg (1994)
13. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
14. Li, Z., Yang, Y., Liu, J., Zhou, X., Lu, H.: Unsupervised feature selection using nonnegative spectral analysis. In: AAAI (2012)
15. Maugis, C., Celeux, G., Martin-Magniette, M.L.: Variable selection for clustering with gaussian mixture models. Biometrics **65**(3), 701–709 (2009)
16. Nene, S.A., Nayar, S.K., Murase, H., et al.: Columbia object image library (coil-20). Tech. rep., Technical Report CUCS-005-96 (1996)
17. Nie, F., Huang, H., Cai, X., Ding, C.H.Q.: Efficient and robust feature selection via joint l2, 1-norms minimization. In: NIPS (2010)
18. Nie, F., Huang, H., Cai, X., Ding, C.H.: Efficient and robust feature selection via joint $\ell 2$, 1-norms minimization. In: NIPS, pp. 1813–1821 (2010)
19. Qian, M., Zhai, C.: Robust unsupervised feature selection. In: IJCAI (2013)

20. Raileanu, L.E., Stoffel, K.: Theoretical comparison between the gini index and information gain criteria. AMAI (2004)
21. Samaria, F.S., Harter, A.C.: Parameterisation of a stochastic model for human face identification. In: IEEE Workshop on Applications of Computer Vision (1994)
22. Strehl, A., Ghosh, J.: Cluster ensembles–a knowledge reuse framework for combining multiple partitions. Journal of Machine Learning Research (JMLR) **3**, 583–617 (2003)
23. Tabakhi, S., Moradi, P., Akhlaghian, F.: An unsupervised feature selection algorithm based on ant colony optimization. Engineering Applications of Artificial Intelligence **32**, 112–123 (2014)
24. Wang, D., Nie, F., Huang, H.: Unsupervised feature selection via unified trace ratio formulation and K-means clustering (TRACK). In: Calders, T., Esposito, F., Hüllermeier, E., Meo, R. (eds.) ECML PKDD 2014, Part III. LNCS, vol. 8726, pp. 306–321. Springer, Heidelberg (2014)
25. Wang, S., Chang, X., Li, X., Sheng, Q.Z., Chen, W.: Multi-task support vector machines for feature selection with shared knowledge discovery. Signal Processing (December 2014)
26. Wang, S., Tang, J., Liu, H.: Embedded unsupervised feature selection. AAAI (2015)
27. Yang, Y., Shen, H.T., Ma, Z., Huang, Z., Zhou, X.: l2, 1-norm regularized discriminative feature selection for unsupervised learning. In: IJCAI (2011)
28. Yang, Y., Zhuang, Y., Wu, F., Pan, Y.: Harmonizing hierarchical manifolds for multimedia document semantics understanding and cross-media retrieval. IEEE Transactions on Multimedia **10**(3), 437–446 (2008)
29. Yang, Y., Ma, Z., Hauptmann, A.G., Sebe, N.: Feature Selection for Multimedia Analysis by Sharing Information Among Multiple Tasks. IEEE TMM **15**(3), 661–669 (2013)
30. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: ICML
31. Zhu, X., Huang, Z., Yang, Y., Shen, H.T., Xu, C., Luo, J.: Self-taught dimensionality reduction on the high-dimensional small-sized data. Pattern Recognition **46**(1), 215–229 (2013)
32. Zhu, X., Suk, H.-I., Shen, D.: Matrix-similarity based loss function and feature selection for alzheimer's disease diagnosis. In: IEEE CVPR, pp. 3089–3096 (2014)
33. Zhu, X., Suk, H.-I., Shen, D.: Discriminative feature selection for multi-class alzheimer's disease classification. In: MLMI, pp. 157–164 (2014)