Joint Clustering and Feature Selection

Liang Du[1,2,3] and Yi-Dong Shen[1]

[1] State Key Laboratory of Computer Science,
Institute of Software, Chinese Academy of Sciences, Beijing, 100190, China
[2] Graduate University of Chinese Academy of Sciences,
[3] University of Chinese Academy of Sciences, Beijing, 100049, China
{duliang, ydshen}@ios.ac.cn

Abstract. Due to the absence of class labels, unsupervised feature selection is much more difficult than supervised feature selection. Traditional unsupervised feature selection algorithms usually select features to preserve the structure of the data set. Inspired from the recent developments on discriminative clustering, we propose in this paper a novel unsupervised feature selection approach via Joint Clustering and Feature Selection (JCFS). Specifically, we integrate Fisher score into the clustering framework. We select those features such that the fisher criterion is maximized and the manifold structure can be best preserved simultaneously. We also discover the connection between JCFS and other clustering and feature selection methods, such as discriminative K-means, JELSR and DCS. Experimental results on real world data sets demonstrated the effectiveness of the proposed algorithm.

Keywords: Unsupervised Feature Selection, Fisher Score, Spectral Clustering

1 Introduction

In many applications in data mining and machine learning, one is often confronted with very high dimensional data. High dimensional data require more time and space to process. Moreover, due to the existence of many irrelevant and/or redundant features, learning algorithms tend to overfit. To overcome this problem, feature selection techniques are designed to select a subset of features from the high dimensional feature set for a compact and accurate data representation.

Feature selection methods can be classified into supervised and unsupervised methods. Supervised feature selection algorithms, such as Fisher score [1], robust sparse regression [2], and trace ratio [3], select features according to the relevance between features and labels. However, in practice, the labels are expensive to obtain. Hence, it is important to develop unsupervised feature selection algorithms by using all the data points without labels.

Recently, several unsupervised algorithms have been proposed to leverage both the manifold structure and learning mechanism. These methods include

Laplacain Score (Lap Socre) [4], Spectral Feature Selection (SPEC) [5], Multi-Cluster Feature Selection (MCFS) [6], Unsupervised Discriminative Feature Selection (UDFS) [7], joint Feature Selection and Subspace Learning (FSSL) [8] and Joint Embedding Learning and Sparse Regression (JELSR) [9]. Commonly, these approaches characterize manifold structure via various graphs and select features with different learning mechanism. LapScore and SPEC rank each features by computing different metrics. MCFS achieves feature selection by Spectral Regression [10]. Both of UDFS, FSSL and JELSR can viewed as an integration of embedding with different graphs and sparse subspace learning via $\ell_{2,1}$-norm regularization.

Besides, to inherit the discriminative power of supervised approach, several recent work [11–13] incorporate supervised subspace learning technique into the clustering framework. Empirical results have shown performance improvement with other popular clustering algorithms.

Based on the above motivation, in this paper, we propose a new approach, called *Jointly Clustering and Feature Selection* (JCFS), for unsupervised feature selection. Specifically, we integrate supervised feature selection technique (Fisher Score) and spectral clustering (manifold learning) in a unified framework. Based on the selected features, we jointly maximize Fisher criterion for feature selection and minimize the spectral clustering criterion to preserve the manifold structure. Compared with traditional unsupervised feature selection approaches, our method could integrate the merits of supervised feature selection and clustering (manifold learning). We also discover the connection between JCFS and other clustering and feature selection methods, such as spectral clustering, discriminative K-means, JELSR and Discriminative codebook selection [14]. Many experimental results are provided for demonstration.

2 Notations

Suppose we have n data points $\{\boldsymbol{x}_i\}_{i=1}^n \subset \mathbf{R}^d$, the data matrix is denoted by $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n] \in \mathbf{R}^{d\times n}$. We use $\boldsymbol{x}^i$ to represent the i-th feature of X. We represent a clustering of the data points by a partition matrix $P = [\boldsymbol{p}_1, \ldots, \boldsymbol{p}_c] = [p_{ij}] \in \{0,1\}^{n\times c}$ and $P\mathbf{1}_c = \mathbf{1}_n$. Thus, exactly one element in each row of P is 1. Instead of directly using the entries of partition matrix P as the cluster labels, we use a Scaled Partition Matrix $Y = P(P^TP)^{-1/2}$. Without loss of generality, we assume that X has been centered with zero mean, i.e., $\sum_i^n \boldsymbol{x}_i = \mathbf{0}$.

3 A Review of Fisher Score and Spectral Clustering

In this section, we briefly introduce Fisher Score [1] and Spectral Clustering [15].

3.1 Fisher Score for Feature Selection

The key idea of Fisher score [1] is to find a subset of features, based on theses selected features the distances between data points in different classes are as large as possible, while the distances between data points in the same class are as small as possible. The *Fisher Score* can be formulated as the following *Ratio Trace* optimization problem:

$$\max_{X^m} \quad \mathrm{tr}(S_t + \gamma I_{d\times d})^{-1} S_b \tag{1}$$

where tr$(\cdot)$ is the trace of a squared matrix, X^m is the data matrix represented by m selected features, the total scatter matrix S_t and the between-cluster scatter matrix S_b are defined as follows [12]:

$$S_t = X^m (X^m)^T \quad S_b = X^m Y Y^T (X^m)^T \tag{2}$$

The regularization parameter $\gamma > 0$ in (1) is used to keep the total scatter matrix non-singular. To represent whether a feature is selected or not, we introduce an indicator variable $\boldsymbol{z} = [z_1, \ldots, z_d]^T$ and $z_i \in \{0,1\}, i = 1, \ldots, d$. Then the Fisher score in (1) can be equivalently reformulated as follows,

$$\begin{aligned} \max_{\boldsymbol{z}} \quad & \mathrm{tr}[(\mathrm{diag}(\boldsymbol{z}) X X^T \mathrm{diag}(\boldsymbol{z}) + \gamma I_{d\times d})^{-1} \mathrm{diag}(\boldsymbol{z}) X Y Y^T X^T \mathrm{diag}(\boldsymbol{z})] \\ \text{s.t.} \quad & \boldsymbol{z} \in \{0,1\}^d, \boldsymbol{z}^T \mathbf{1}_d = m. \end{aligned} \tag{3}$$

where diag$(\boldsymbol{z})$ is a diagonal matrix whose diagonal elements are z_i. The objective function in (3) can be further simplified as follows:

$$\begin{aligned} \mathcal{L}(\boldsymbol{z}) &= \mathrm{tr}[Y^T X^T \mathrm{diag}(\boldsymbol{z})(\mathrm{diag}(\boldsymbol{z}) X X^T \mathrm{diag}(\boldsymbol{z}) + \gamma I_{d\times d})^{-1} \mathrm{diag}(\boldsymbol{z}) X Y] \\ &= \mathrm{tr}[Y^T X^T \mathrm{diag}(\boldsymbol{z}) X (X^T \mathrm{diag}(\boldsymbol{z}) X + \gamma I_{n\times n})^{-1} Y] \\ &= \mathrm{tr}[Y^T (I_{n\times n} - \gamma (X^T \mathrm{diag}(\boldsymbol{z}) X + \gamma I_{n\times n})^{-1}) Y] \end{aligned} \tag{4}$$

It is easy to verify that $Y^T Y = I$. Therefore, it can be further simplified as

$$\min_{\boldsymbol{z}} \quad \mathrm{tr}[Y^T (X^T \mathrm{diag}(\boldsymbol{z}) X + \gamma I_{n\times n})^{-1}) Y] \quad \text{s.t.} \quad \boldsymbol{z} \in \{0,1\}^d, \boldsymbol{z}^T \mathbf{1}_d = m. \tag{5}$$

3.2 Spectral Clustering

Spectral clustering is one of the most popular clustering methods in recent years, which exploits the eigen-structure of a specially constructed matrix. Generally, the objective function of spectral clustering [15] algorithms can be defined as:

$$\min_{Y} \quad \mathrm{tr}(Y^T L Y) \quad \text{s.t.} \quad Y^T Y = I \tag{6}$$

where L is a Laplacian matrix to approximate the manifold structure with different choices. In order to capture the local data structure, one can construct a nearest neighbor graph with an affinity matrix $A \in \mathcal{R}^{n\times n}$. The simplest definition of A is as follows:

$$A_{ij} = \begin{cases} 1, \boldsymbol{x}_i \in \mathcal{N}(\boldsymbol{x}_j) \text{ or } \boldsymbol{x}_j \in \mathcal{N}(\boldsymbol{x}_i) \\ 0, \text{otherwise.} \end{cases} \tag{7}$$

The normalized Laplacian matrix L is then defined by $L = I - D^{-1/2}AD^{-1/2}$, where D is a diagonal matrix with the diagonal elements as $D_{ii} = \sum_j^n A_{ij}, \forall i$. The optimization problem in (6) can be solved by the eigenvalue decomposition. Based on the continuous solution, the final discrete solution is then obtained by K-means or spectral rotation [15].

4 Proposed Formulation

In this section, we will integrate Fisher score and spectral clustering in a unified framework. The key idea of our method is to find a subset of features, based on which we jointly maximize the Fisher criterion for feature selection and minimize the spectral clustering criterion to best preserve the manifold structure. It can be mathematically formulated as follows,

$$\begin{aligned} \min_{\boldsymbol{Y},\boldsymbol{z}} \quad & \mathrm{tr}(Y^T(L + \lambda(X^T \mathrm{diag}(\boldsymbol{z})X + \gamma I)^{-1})Y) \\ \text{s.t.} \quad & Y^TY = I, \boldsymbol{z} \in \{0,1\}^d, \boldsymbol{z}^T\mathbf{1}_d = m, \end{aligned} \tag{8}$$

where λ and γ are two tradeoff parameters. Due to the integer constraints on $\boldsymbol{z}$ problem (8) is a mixed integer programming [16].

4.1 Optimization of JCFS Algorithm

In this section, we propose to develop an iterative algorithm to solve the optimization problem in (8).

The Computation of $\boldsymbol{Y}$ for Given $\boldsymbol{z}$

$$\min_{Y} \quad \mathrm{tr}(Y^T(L + \lambda(X^T \mathrm{diag}(\boldsymbol{z})X + \gamma I)^{-1})Y) \quad \text{s.t.} \quad Y^TY = I. \tag{9}$$

Though a matrix inverse is involved, we will show that the inverse can be efficiently computed without explicit inverse operation. The above optimization problem integrates Laplacian matrix and inverse covariance matrix based on s-elected features in a unified framework, which can be solved by the eigenvalue decomposition.

The Computation of $\boldsymbol{z}$ for Given $\boldsymbol{Y}$ Given an existing Y, the above optimizing with respect to $\boldsymbol{z}$ is equivalent to solving the following minimization problem

$$\min_{\boldsymbol{z}} \quad \mathrm{tr}(Y^T(X^T \mathrm{diag}(\boldsymbol{z})X + \gamma I)^{-1}Y) \quad \text{s.t.} \quad \boldsymbol{z} \in \{0,1\}^d, \boldsymbol{z}^T\mathbf{1}_d = m. \tag{10}$$

The above problem is NP-hard due to the combination nature. To solve it efficiently, we can relax the integer constraints on $\boldsymbol{z}$ using nonnegative ℓ_1-norm regularization. The following relaxed problem is a convex optimization problem which can be solved efficiently.

$$\min_{\boldsymbol{z}} \quad \mathrm{tr}(Y^T(\sum_{i=1}^{d} z_i \boldsymbol{x}^i(\boldsymbol{x}^i)^T + \gamma I)^{-1}Y) + \mu \boldsymbol{z}^T \mathbf{1}_d \quad \text{s.t.} \quad \boldsymbol{z} \geq 0 \tag{11}$$

where $\mu > 0$ is a regularization parameter to control the sparseness of $\boldsymbol{z}$. Note that (11) is no longer equivalent to (10) due to the relaxation. In other word-s, the relaxation can be seen as a tradeoff between the strict equivalence and computational tractability.

Remember that our goal is to selected m features, we did not solve the above convex optimization problem in this paper. Instead, we solve a sequential optimization problem by specifying the number of selected features m, which is particularly convenient for feature selection. Suppose the first t features have been selected, i.e., $\boldsymbol{x}^1, \ldots, \boldsymbol{x}^t$, then the $(t+1)$-th feature can be obtained by solving the following problem:

$$\boldsymbol{x}^{t+1} = \operatorname*{argmin}_{\boldsymbol{x}^j} \quad \mathrm{tr}(Y^T(\sum_{i=1}^{t} \boldsymbol{x}^i(\boldsymbol{x}^i)^T + \gamma I + \boldsymbol{x}^j(\boldsymbol{x}^j)^T)^{-1}Y) \tag{12}$$

By using the Woodbury-Morrison formula [17], we have

$$(\sum_{i=1}^{t} \boldsymbol{x}^i(\boldsymbol{x}^i)^T + \gamma I + \boldsymbol{x}^j(\boldsymbol{x}^j)^T)^{-1} = M_t - \frac{M_t \boldsymbol{x}^j(\boldsymbol{x}^j)^T M_t}{1 + (\boldsymbol{x}^j)^T M_t \boldsymbol{x}^j}. \tag{13}$$

where $M_t = (\sum_{i=1}^{t} \boldsymbol{x}^i(\boldsymbol{x}^i)^T + \gamma I)^{-1}$. Eq. (13) can be used to compute the inverse of M_{t+1} efficiently without inverse operation. Plug (13) into (12), it is equivalent to solving the following problem

$$\boldsymbol{x}^{t+1} = \operatorname*{argmax}_{\boldsymbol{x}^j} \quad \frac{\boldsymbol{x}^j M_t Y Y^T M_t \boldsymbol{x}^j}{1 + (\boldsymbol{x}^j)^T M_t \boldsymbol{x}^j} \tag{14}$$

which is computed on each feature independently. The above process is repeated until we have selected m features. More details about the sequential optimization scheme can be found at [14, 18]. We summarize the complete JCFS algorithm for feature selection in Algorithm (1).

5 Connection to prior Work

In this section, we discuss the connection between JCFS and SEC, Discriminative K-means, JELSR [9] and DCS [14]. The first two are proposed for clustering task, while the last two are designed for unsupervised feature selection.

5.1 Connection between JCFS and Spectral Clustering

JCFS reduces to spectral clustering, if λ is set as zero. Therefore spectral clustering is a special case of JCFS.

Algorithm 1 JCFS for Feature Selection

```
Input: data set $X$, The number of clusters $c$, The number of selected features $m$, The
    number of nearest neighbors $k$, parameters $\lambda$ and $\gamma$
Output: $m$ selected features.
Construct the Laplacian matrix $L$ on a nearest neighborhood graph;
repeat
    Solve (9) by the eigenvalue decomposition and obtain the optimal $Y$;
    Set $t = 0$ and initialize $M_0 = \frac{1}{\gamma}I$;
    for $t = 1$ to $m$ do
        Select $t$-th feature according to (14);
        Compute $M_t$ according to (13);
    end for
until Selected features in consecutive iterations are not change
```

5.2 Connection between JCFS and Discriminative K-means

Several recent work [11, 12] incorporate supervised dimension reduction such as Linear Discriminant Analysis (LDA) [19] into the clustering framework, which jointly learn the low-dimensional subspace and clustering. For instance, Discriminative Clustering methods solve the following optimization problem:

$$\max_{W,Y} \quad \mathrm{tr}((W^T(S_t + \gamma I)^{-1}W)(W^T S_b W)) \tag{15}$$

There are two sets of variables, the projection matrix W and the scaled cluster assignment matrix Y, in (15). The above optimization problem can be simplified by optimizing Y only based on the following observation [12]:

$$\mathrm{tr}((W^T(S_t + \gamma I)^{-1}W)(W^T S_b W)) \leq \mathrm{tr}((S_t + \gamma I)^{-1} S_b) \tag{16}$$

where the equality holds when $W = VS$, and V is composed of the eigenvectors of $(S_t + \gamma I)^{-1} S_b$ corresponding to all the nonzero eigenvalues, S is an arbitrary nonsingular matrix. Based on (16), JCFS reduces to Discriminant K-means, if $\lambda \to \infty$ and $m = d$. Therefore Discriminant K-means is a special case of JCFS.

5.3 Connection between JCFS and JELSR

JELSR [9] integrate embedding learning and sparse regression in a unified framework to perform unsupervised feature selection. It can be formulated as follows.

$$\min_{W,Y} \quad \mathrm{tr}(Y^T L Y) + \beta(||X^T W - Y||_2^2 + \alpha||W||_{2,1}) \quad \text{s.t.} \quad Y^T Y = I \tag{17}$$

The sparse subspace W and embedding Y in (17) are learned iteratively. To analysis the connection between JCFS and JELSR, we follow the updating rules of JELSR. Suppose the t-th iteration of W is given by W_t, the $(t+1)$-th iteration can be obtained by $W_{t+1} = (XX^T + \alpha U)^{-1} XY$, where U is a diagonal matrix and $U_{ii} = \frac{1}{||2(W_t)_i||_2}$. Substitute W_{t+1} into (17), the optimization of Y leads to

$$\min_{Y} \quad \mathrm{tr}(Y^T(L + \beta I_{n\times n} - \beta X^T(XX^T + \alpha U)^{-1}X)Y) \quad \text{s.t.} \quad Y^T Y = I \tag{18}$$

which can be further reformulated into

$$\min_{Y} \quad \mathrm{tr}(Y^T(L+\beta\alpha(X^T\mathrm{diag}(z)X+\alpha I)^{-1})Y) \quad \text{s.t.} \quad Y^TY=I. \tag{19}$$

where $z_i = ||\boldsymbol{w}_i||_2$. When Y is fixed the optimization of W is solved by the following regularized problem,

$$\min_{W} \quad ||X^TW-Y||_2^2+\alpha||W||_{2,1} \tag{20}$$

It is mathematically similar to Group Lasso problem [2] and multi-task feature selection [20]. Based on above formulation, we found that JELSR and JCFS solve similar problem to update Y, the difference lies JCFS only use selected features (0-1 weights of features) while JELSR use all features weighted by its norm $||\boldsymbol{w}_i||_2$. For feature selection, JELSR solve a $\ell_{2,1}$-norm minimization problem which leads to sparsity of W, however JCFS directly select features to maximizes Fisher score.

Besides, the differences between JCFS and MRSF [21], MCFS [6] can also be obtained based on the above analysis. MRSF and MCFS are two-stage approaches, which both first learn the graph embedding. MRSF then selects features by solving the optimization problem (20), while MCFS selects features via solving K independently LASSO problems.

5.4 Connection between JCFS and Discriminative Codeword Selection

DCS [14] was proposed to minimize the fitting error based on selected features, which can be formulated as follow.

$$\min_{W,b,\boldsymbol{z}} ||Y-X^T\mathrm{diag}(\boldsymbol{z})W-1_mb^T||_F^2+\alpha||W||_F^2 \tag{21}$$

After substituting the optimal value of W and b into (21), the above optimization problem is equivalent to

$$\min_{Y,z} \quad \mathrm{tr}(Y'H_n(\sum_{i=1}^{d} z_i\boldsymbol{x}^i(\boldsymbol{x}^i)^T+\alpha I)^{-1}H_nY) \tag{22}$$

$$\text{s.t.} \quad Y\in\{0,1\}^{n\times r}, Y\mathbf{1}_r=\mathbf{1}_n, \boldsymbol{z}\in\{0,1\}^n, \mathbf{1}_n^T\boldsymbol{z}=m, \tag{23}$$

where $H_n = I_{n\times n}-\frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T$ is the centering matrix.

It is clear that the above formulation is essentially the same as the discriminative feature selection part in JCFS, however we derive JCFS from supervised feature selection (*Fisher Score*) criterion, while DCS is derived from *ridge regression.* Besides JCFS also integrate spectral clustering to capture manifold information which have been shown to be useful for unsupervised feature selection [4, 6].

6 Experiments

In this section, several experiments are carried out to show the effectiveness of our proposed JCFS for feature selection. We perform clustering and nearest neighbor classification experiments by only using the selected features. The following five unsupervised feature selection algorithms are compared:

- Max Variance which selects those features of maximum variances in order to obtain the best expressive power.
- Laplacian Score [4] which selects the features most consistent with the Gaussian Laplacian matrix.
- Multi-Cluster Feature Selection (MCFS)[4] [6] which selects features using spectral regression with ℓ_1-norm regularization.
- JELSR [9] which performs feature selection via unifying the graph embedding and sparse regression.
- Our proposed JCFS algorithm.

6.1 Data Sets

We use four real world data sets in our experiments, the processed version of these data sets can be obtained from Cai's page[5].

The first one is the ORL face database which consists of a total of 400 face images, of a total of 40 subjects (10 samples per subject). The size of each cropped image is 32 × 32 pixels, with 256 gray levels per pixel. Thus, each face image can be represented by a 1,024-dimensional vector.

The second one is the COIL image library from Columbia. It contains 20 objects. The images of each objects were taken 5 degrees apart as the object is rotated on a turntable and each objects has 72 images. The size of each image is 32 × 32 pixels, with 256 gray levels per pixel. Thus, each image is represented by a 1,024-dimensional vector.

The third one is Isolet spoken letter recognition data[6]. This data set was generated as follows. 150 subjects spoke the name of each letter of the alphabet twice. The speakers are grouped into sets of 30 speakers each, and are referred to as isolet1 through isolet5. In our experiments, we use isolet5 which consists 1559 examples with 617 features.

The fourth one is Extended Yale-B database[7] contains 16128 face images of 38 human subjects under 9 pose and 64 illumination conditions. In our experiment, we choose the frontal pose and use all the images under different illumination, thus we get 2414 image in total. They are resized to 32 × 32 pixels, with 256 gray levels per pixel. Thus each face image is represented as a 1024-dimensional vector.

[4] The code is available at `http://www.zjucadcg.cn/dengcai/MCFS/index.html`. The code for Laplacian Score can also be found in this page.

[5] `http://www.zjucadcg.cn/dengcai/Data/data.html`

[6] `http://archive.ics.uci.edu/ml/datasets/ISOLET`

[7] `http://vision.ucsd.edu/leekc/ExtYaleDatabase/ExtYaleB.html`

6.2 Clustering

We perform K-means clustering by using the selected features and compare the results of different algorithms in this test.

Evaluation Metrics The performance is evaluated by comparing the labels obtained using clustering algorithms with the ground truth labels. We use Clustering Accuracy (ACC) and Normalized Mutual Information (NMI) to measure the clustering performance.

Parameter Settings Several parameters need to be set beforehand for these algorithms. All the compared algorithms except Max Variance are need to determine the parameter k which specifies the number of nearest neighbors. Generally speaking, k should be set as a small number to preserve the local manifold structure. To fairly compare their performances, we fixe k as 5, used in [6, 7], for all the algorithms on all the data sets. For Lap Socre, MCFS, and JCFS we use binary weights to represent the nearest graph for its simplicity. The dimensionality of graph embedding in MCFS and JELSR is fixed as the number of clusters. For JELSR, we fix $\alpha = 1$ and search regularization parameter β from the $\{10^{-4}, 10^{-3}, 10^{-2}, 0.1, 1\}$. For our JCFS, we simply set $\gamma = 10^{-4}$ and search the best parameter of λ from the grid $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}\}$. The total number of clusters c is provided for all the clustering algorithms. To reduce the statistical variation of K-means results, we independently repeat the experiments for 100 times with random initializations. We report the best average results corresponding to the best objective values in terms of ACC and NMI respectively.

Clustering Results Table (1, 2 , 3, and 4) show the clustering performance versus the number of selected features m on different data sets. As can be seen, JCFS performs better than the other methods in most cases. Comparing with second best method, JCFS achieves 8.1% (4.2%), 14.1% (10.1%), 4.9% (7.4%), and 8.8% (8.4%) relative improvements in average when measured by ACC (N-MI) on the ORL, COIL20, ISOLET, YABLEB data sets, respectively. This show that the idea of integrating supervised feature selection into the clustering framework is beneficial in designing unsupervised feature selection methods. It would be interesting to note that, on these data sets, our proposed algorithm performs surprisingly well by using only very few features, such as 5 and 15 features.

6.3 Nearest Neighbor Classification

In this subsection, we evaluate different feature selection algorithms in the classification task. We perform nearest neighbor classifier (1-NN) with the selected features. Classification accuracy is used to measure the performance. The experimental results are shown in Table 5, we can observe that JCFS is competitive with other algorithms for nearest neighbor classification different selected features m.

Table 1. Clustering performance on the ORL with m selected features

m	ACC					NMI				
	MaxVar	LapScore	MCFS	JELSR	JCFS	MaxVar	LapScore	MCFS	JELSR	JCFS
5	29.4	38.7	36.2	37.2	**43.1**	54.5	62.2	59.8	60.4	**65.9**
15	31.1	40.4	47.3	43.3	**51.1**	56.4	64.3	70.2	66.4	**72.8**
25	33.2	41.8	50.6	46.7	**52.6**	58.5	65.4	72.6	69.3	**74.1**
35	35.1	41.5	49.6	47.8	**53.7**	60.3	65.6	71.6	70.9	**74.9**
50	37.2	44.9	51.5	49.2	**53.7**	61.8	68.6	74.1	71.9	**75.0**

Table 2. Clustering performance on the COIL with m selected features

m	ACC					NMI				
	MaxVar	LapScore	MCFS	JELSR	JCFS	MaxVar	LapScore	MCFS	JELSR	JCFS
5	33.9	33.4	45.5	43.4	**53.0**	51.0	50.6	55.3	55.8	**63.0**
15	40.1	41.1	49.2	53.2	**62.5**	58.1	59.3	63.3	64.5	**73.2**
25	45.2	43.4	51.0	53.3	**61.2**	60.7	62.4	66.2	65.9	**74.5**
35	46.4	50.1	54.8	54.7	**60.5**	62.5	65.3	69.7	68.2	**74.9**
50	47.4	52.8	53.9	56.1	**60.2**	65.2	67.1	69.6	70.8	**74.7**

Table 3. Clustering performance on the ISOLET with m selected features

m	ACC					NMI				
	MaxVar	LapScore	MCFS	JELSR	JCFS	MaxVar	LapScore	MCFS	JELSR	JCFS
5	20.3	18.9	26.7	**33.1**	31.2	29.3	39.5	37.6	**50.3**	49.4
15	26.4	32.8	46.7	43.4	**54.2**	38.9	50.6	63.0	59.6	**69.6**
25	37.2	35.6	52.0	46.8	**52.0**	51.9	54.2	68.6	63.2	**69.6**
35	38.0	41.9	**51.8**	46.5	51.0	54.9	61.8	67.6	65.1	**69.4**
50	39.7	43.4	53.1	52.1	**53.3**	59.2	63.4	70.2	68.4	**71.9**

Table 4. Clustering performance on the YALEB with m selected features

m	ACC					NMI				
	MaxVar	LapScore	MCFS	JELSR	JCFS	MaxVar	LapScore	MCFS	JELSR	JCFS
5	8.9	8.7	15.0	11.0	**17.0**	13.9	13.8	25.2	17.9	**28.0**
15	9.2	8.6	14.1	10.5	**15.0**	15.2	13.6	22.1	18.2	**24.0**
25	9.2	8.8	14.9	10.3	**15.2**	13.8	14.1	**24.7**	17.1	23.3
35	9.2	8.8	13.6	10.2	**15.1**	13.2	13.9	20.7	16.7	**23.3**
50	9.1	8.8	13.3	10.1	**14.9**	13.2	13.8	19.7	16.1	**23.4**

Table 5. 1-NN classification accuracy on all the data sets with m selected features

DataSet	ORL			COIL			ISOLET			YALEB		
m	5	25	50	5	25	50	5	25	50	5	25	50
MaxVar	40.3	62.5	71.3	55.8	78.8	84.5	20.3	57.6	70.5	17.8	31.5	35.7
LapScore	51.2	71.8	83.0	55.8	74.5	83.3	21.0	54.1	73.6	10.4	16.9	20.0
MCFS	49.5	**92.0**	93.5	80.8	96.4	98.8	26.4	71.6	78.8	30.9	**59.5**	60.9
JELSR	50.7	85.5	93.3	70.3	96.1	99.2	**38.7**	71.8	78.3	21.7	44.5	53.0
JCFS	**63.2**	91.3	**94.0**	**82.7**	**99.3**	**99.9**	36.1	**75.5**	**80.8**	**40.3**	55.6	**64.7**

6.4 Parameters Selection

Though our algorithms have three parameters, that are, regularization parameters (λ and γ) and the number of nearest neighbors k. The parameter k is commonly used for manifold learning and spectral clustering, and the results are often stable when k is small. The parameter γ can be safely set as a small number. Due to space limitation, we only show the clustering results with different λ for different selected features m. The data set used for this test is the ORL face database. The experimental results are shown in Fig 1

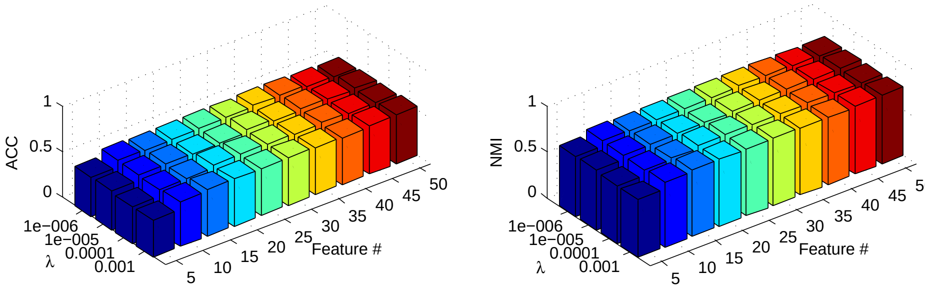

Fig. 1. Clustering performance variation of JCFS on ORL data set w.r.t different λ

7 Conclusions and Future Work

In this paper, we have proposed a new unsupervised feature selection approach which selected features that jointly maximize the supervised Fisher criterion and best preserve the manifold information. We show that spectral clustering, discriminative K-means, DCS are all the special cases of JCFS. We also prove that JELSR and JCFS share similar objective function, where features in JELSR are weighted by its norm while JCFS use a 0-1 weight scheme. Our preliminary results on clustering and classification with selected features show that JCFS outperforms the compared algorithms. In future, we plan to explore other supervised feature selection techniques into the clustering framework for unsupervised feature selection.

8 Acknowledgments

We would like to thank all anonymous reviewers for their helpful comments. This work is supported in part by NSFC grant 60970045 and China National 973 project 2013CB329305.

References

1. Duda, R., Hart, P., Stork, D.: Pattern Classification (2nd Edition). Volume 2. (2001)
2. Nie, F., Huang, H., Cai, X., Ding, C.: Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization. Advances in Neural Information Processing Systems (NIPS) (2010)
3. Nie, F., Xiang, S., Jia, Y., Zhang, C., Yan, S.: Trace ratio criterion for feature selection. In: Proceedings of the 23rd international conference on Artificial intelligence (AAAI). Volume 2. (2008) 671–676
4. He, X., Cai, D., Niyogi, P.: Laplacian score for feature selection. Advances in Neural Information Processing Systems (NIPS) **18** (2006) 507–514
5. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: Proceedings of the 24th ICML. (2007) 1151–1157
6. Cai, D., Zhang, C., He, X.: Unsupervised feature selection for multi-cluster data. In: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD). (2010) 333–342
7. Yang, Y., Shen, H.T., Ma, Z., Huang, Z., Zhou, X.: ℓ_{21}-norm regularized discriminative feature selection for unsupervised learning. In: Proceedings of the 22th IJCAI. (2011) 1589–1594
8. Gu, Q., Li, Z., Han, J.: Joint feature selection and subspace learning. In: Proceedings of the 22th IJCAI. (2011) 1294–1299
9. Hou, C., Nie, F., Yi, D., Wu, Y.: Feature selection via joint embedding learning and sparse regression. In: Proceedings of the 22th IJCAI. (2011) 1324–1329
10. Cai, D., He, X., Han, J.: Spectral regression: A unified approach for sparse subspace learning. In: Seventh IEEE International Conference on Data Mining (ICDM). (2007) 73–82
11. Ding, C., Li, T.: Adaptive dimension reduction using discriminant analysis and k-means clustering. In: Proceedings of the 24th ICML. (2007) 521–528
12. Ye, J., Zhao, Z., Wu, M.: Discriminative k-means for clustering. Advances in Neural Information Processing Systems (NIPS) **20** (2007) 1649–1656
13. Bach, F., Harchaoui, Z.: Diffrac: A discriminative and flexible framework for clustering. Advances in Neural Information Processing Systems (NIPS) **20** (2008) 49–56
14. Zhang, L., Chen, C., Bu, J., Chen, Z., Tan, S., He, X.: Discriminative codeword selection for image representation. In: Proceedings of the international conference on Multimedia, ACM (2010) 173–182
15. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **22**(8) (2000) 888–905
16. Boyd, S., Vandenberghe, L.: Convex Optimization. (2004)
17. Strang, G.: Introduction to Linear Algebra. Wellesley Cambridge Pr (2003)
18. He, X., Ji, M., Zhang, C., Bao, H.: A variance minimization criterion to feature selection using laplacian regularization. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) **33**(10) (2011) 2013–2025
19. Fukunaga, K.: Introduction to Statistical Pattern Recognition (2nd ed.). (1990)
20. Liu, J., Ji, S., Ye, J.: Multi-task feature learning via efficient l 2, 1-norm minimization. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, AUAI Press (2009) 339–348
21. Zhao, Z., Wang, L., Liu, H.: Efficient spectral feature selection with minimum redundancy. In: Proceedings of the 24th AAAI. (2010)