Robust Feature Selection via L21-Norm Minimization and Maximization

First A. Xu Ma, Second B. Qaiolin Ye, IEEE Member, Third C. Shangbing Gao, IEEE Member, Third C. Author, He Yan

***Abstract*— Feature selection and feature transformation are two different methods for dimensionality reduction. However, they are always presented separately. Feature transformation aims to find a new feature subspace while feature selection devotes to selecting a subset of original feature set. To obtain a better method for dimensionality reduction, we proposed a new feature selection method which fuses the feature transformation method Linear Discriminant Analysis (LDA) based on L21-norm. The new formulation is more likely to support better robustness. It is challenging to solve the resulted objective because it minimizes and maximizes non-smooth L21-norm terms simultaneously. As an important work of this paper, we proposed an iterative algorithm to solve this problem. A series of theories prove that this algorithm is convergent and computationally efficient. Experimental results on various data sets demonstrate the effectiveness of our new method.**

***Index Terms*— Feature selection, L21-norm, Linear discriminant analysis (LDA)**

I. INTRODUCTION

IN many applications of data mining and pattern recognition, data are characterized by high dimensionality. Too much features increase the computational time and space for processing data. Moreover, lots of features are redundant and irrelevant, which do not contribute to classification[1-3]. Hence, dimensionality reduction has been a vital part of data processing in the field of pattern recognition. Dimensionality reduction is committed to finding the "intrinsic dimensionality" [4, 5] of the data. This leads one to consider a method that prune the useless features or represent the input data in a lower dimensional space.

Dimensionality reduction[6-9] can be divided into two ways: feature transformation and feature selection. Feature transformation methods transform the original features to a new subspace with lower dimensionality. Different from feature transformation, feature selection tries to erase the irrelevant or redundant features and reserve the most discriminative features simultaneously. Consequently, feature selection[1, 2, 10] keeps the primary semantics of features, and provides an interpretability for the new features. As a result, feature selection has been more and more inviting and many studies focus on feature selection in recent years. In recent research, more and more people pay attention to the combination of feature transformation and feature selection [11-13].

Fisher Linear Discriminant Analysis (LDA)[14-17] is one of the most popular supervised feature transformation methods. LDA searches for a new subspace, which maximizes the "between classes scatter" and minimizes the "with-in classes scatter" simultaneously. This criterion allows different classes of data points to be separated as much as possible and the same classes of data points are gathered as much as possible in the new projected space. In the past decades, a number of extensions to LDA were developed to convert it into feature selection methods[18-22]. The Fisher Score algorithm[23] is a popular feature selection method base on linear discriminant analysis. This method values and ranks each feature separately by computing the variance between the feature and the same type of samples, and then selects the top ranked features as the objective features. However, this method ignores the relationship between features and overpasses the existence of redundant features. Consequently, Fisher Score does not have the ability to remove feature redundancy, and can not deal with feature interaction. In order to overcome this shortcoming, M. Masaeli *et al.* propose a new algorithm named Linear Discriminant Feature Selection (LDFS)[24]. It is a filter-based feature selection method which is inspired by LDA. LDFS equips the traditional LDA with $l_{\infty 1} - norm$ regularization terms [25, 26]. Since the selected features are obtained by the learning mechanism, LDFS can remove the redundant features and irrelevant features simultaneously. Thus LDFS plays an important role in feature selection. However, the formulation of LDFS is defective because it ignores the possibility of the arbitrary scalability of the transformation matrix. The arbitrary scalability can lead to a trivial solution of all zero. Thus, LDFS cannot obtain the most discriminative features when arriving at the trivial solution of zero.

In 2016, Hong Tao *et al.* propound a new formulation named as Discriminative Feature Selection (DFS)[18], which can void the trivial solution by constraining the formulation to be unrelated. Instead of solving the minimization term and maximization term simultaneously, DFS enforces a term in the

The work is supported in part by the National Science Foundation of China under Grants 61401214, 6177320, and 61772275, the Natural Science Foundation of Jiangsu Province under Grants BK20171453, the Jiangsu Key Laboratory for Internet of Things and Mobile Internet Technology, and the Jiangsu Key Laboratory of Image and Video Understanding for Social Safety. (*Corresponding author: Q. Ye*)

X. Ma and Q. Ye with the Laboratory for Internet of Things and Mobile Internet Technology of Jiangsu Province, Huaiyin Institute of Technology, Huaian, 223003, P.R. China and also with College of Information Science and Technology, Nanjing Forestry University, Nanjing, Jiangsu 210037, P. R. China.

S. Gao is with the Laboratory for Internet of Things and Mobile Internet Technology of Jiangsu Province, Huaiyin Institute of Technology, Huaian, 223003, P.R. China.

formulation to be a constraint. Also, the $l_{21}-norm$ regularization [27-30] is introduced in the formulation. These improvements make DFS not only have the advantage of LDFS to remove redundant and irrelevant features simultaneously, but also avoid trivial solution. While DFS is an efficient and creative feature selection method, it is based on the l_2-norm. The l_2-norm distance metric can cause DFS to prone to both outlier data samples and outlier features. In other words, the selected features of DFS may be not the most discriminant features because the learning process could be highly influenced by outlying data samples and features.

In this paper, we address this issue of robustness in the existence of both outlier data points and outlier features. Many previous studies have carried on to improve the robustness of pattern recognition methods by using the $l_{21}-norm$ regularization terms. Thus far, to our best knowledge, there is few works using $l_{21}-norm$ distance metric in the learning formulations for feature selection. Inspired by DFS, we propose such linear discriminant analysis method based on $l_{21}-norm$ distance metric, termed as L21FS, as a robust and discriminative feature selection method. The new L21FS method minimizes and maximizes $l_{21}-norm$ terms simultaneously. The paper is interesting from these contributions as follows.

1. Develop a new formulation based on $l_{21}-norm$ distance metric and solve the maximization problem and minimization problem simultaneously.
2. This new method provides a robust alternative to traditional DFS.
3. An efficient iterative algorithm is proposed for the resulted formulation, and a series of theories prove that this algorithm is strictly convergent and computationally feasible.
4. A large number of experiments on various data sets has proven the ability and robustness of the new method. Furthermore, the new method outperformed the other related state-of-the-art feature selection method.

The remaining content of this paper is organized as follows. In Section 2, we briefly introduce the related work and the notations. Section 3 includes our new method and the theoretical justification. The experimental results are displayed in Section 4. In section 5, we summarize the paper.

II. Related Work

In this section, we introduce the notations and the definitions related in this paper. LDA is a popular dimensionality reduction method in the field of pattern recognition. Suppose we have n data points $\{x_1, x_2 \cdots x_n\}$ belonging to c classes. For convenience, the data set can be presented by matrix X. Besides, X_i denotes the ith data point and X_j denotes the jth feature. The propose of LDA is to seek a projection plane so that the distance between the points of different classes is maximized, the distance between the points of the same class is minimized. To evaluate the distance of data points, the scatter matrix based on l_2-norm is introduced:

$$\mathbf{S}_b=\sum_{k=1}^{c} n^k\left(\mu^k-\mu\right)\left(\mu^k-\mu\right)^T \tag{1}$$

$$\mathbf{S}_w=\sum_{k=1}^{c}\left(\sum_{i=1}^{n_k}\left(x_i^k-\mu^k\right)\left(x_i^k-\mu^k\right)^T\right) \tag{2}$$

$$\mathbf{S}_t=\sum_{i=1}^{n}\left(x_i-\mu\right)\left(x_i-\mu\right)^T \tag{3}$$

where S_b , S_w , S_t mean between-class scatter matrix, within-class scatter matrix and total scatter matrix. The related notations in this paper are give in Table 1.

TABLE I
NOTATIONS

Notation	Description
$\boldsymbol{c}$	The classes number
$\boldsymbol{n}$	Data points number
$\boldsymbol{\mu}$	The mean point of total samples
$\boldsymbol{k}$	The k-th class
$\boldsymbol{S_b}$	The between-class scatter matrix
$\boldsymbol{S_w}$	The within-class scatter matrix
$\boldsymbol{S_t}$	The total scatter matrix
$\boldsymbol{W}$	The projection matrix

It is clear that S_t is the sum of S_b and S_w, which can be written as:

$$\mathbf{S}_t=\mathbf{S}_b+\mathbf{S}_w. \tag{4}$$

The objective of LDA is as following:

$$J(\mathbf{W})=\max_{\mathbf{W}}\frac{\mathbf{W}^T\mathbf{S}_b\mathbf{W}}{\mathbf{W}^T\mathbf{S}_W\mathbf{W}} \tag{5}$$

where W is the projective plane that we seek.

Since the between-class scatter matrix, within-class scatter matrix and total scatter matrix are closely related, the original LDA derives many variations. Also, turning the maximization problem into minimization problem greatly expands this possibility. Inspired of this view, LDFS rewrites the formulation of traditional LDA as:

$$J(\mathbf{W})=\min_{\mathbf{W}}-\frac{\mathbf{W}^T\mathbf{S}_b\mathbf{W}}{\mathbf{W}^T\mathbf{S}_W\mathbf{W}}. \tag{6}$$

For the projection matrix W, each row of it represents the importance of the corresponding feature. If the row is mainly composed of zero, it means that the corresponding feature makes no contribution to the classification. On the contrary, the row of W, which corresponding to the selected feature, must have a nonzero item at least. Consequently, to achieve the ability of feature selection, LDFS must force the projective matrix W to contain more zero rows. Hence, LDFS introduces the $l_{\infty,1}-norm$ regularization term, which is helpful to alleviate over-fitting and improve the generalization

performance. The regularized objective can be written as follow:

$$J(\mathbf{W}) = \min_{\mathbf{W}} -\frac{\mathbf{W}^T\mathbf{S}_b\mathbf{W}}{\mathbf{W}^T\mathbf{S}_W\mathbf{W}} + \gamma\left\|\mathbf{W}\right\|_{\infty,1} = \min_{\mathbf{W}} -\frac{\mathbf{W}^T\mathbf{S}_b\mathbf{W}}{\mathbf{W}^T\mathbf{S}_W\mathbf{W}} + \gamma\sum_{j=1}^{d}\left\|\mathbf{W}_j\right\|_{\infty}. \quad (7)$$

However, the formulation of LDFS would lose the ability of feature selection when arriving at the trivial solution, which is proved in [18]. In 2016, an advanced method based on LDFS has been presented, which is called Discriminative Feature Selection (DFS). In order to avoid arbitrary scaling as well as the trivial solutions, DFS enforces the transformation matrix W to be independent of the S_b . Also, the $l_{2,1}-norm$ regularization is employed instead of $l_{\infty,1}-norm$ regularization. The resulted formulation is as follow:

$$\min_{\mathbf{W}^T\mathbf{S}_t\mathbf{W}=\mathbf{I}} -tr\left(\mathbf{W}^T\mathbf{S}_b\mathbf{W}\right)+\gamma\left\|\mathbf{W}\right\|_{2,1}. \quad (8)$$

The criterion (8) is to minimize the value of between class scatter, and the second term of (8) can balance the sparsity and the experience risk. Since the $l_{\infty,1}-norm$ and $l_{2,1}-norm$ are both extensions of l_1-norm, DFS also employed the $l_{2,p}-norm$ regularization instead of the $l_{\infty,1}-norm$ regularization.

As expounded above, DFS is a better alternative to LDFS due to the handing of trivial solutions. Nevertheless, it still ignores the robustness of feature selection. DFS can not deal with the problem of robustness very well in the existence of both outlier data points and outlier features. Recall the formulation of DFS, it can be rewritten as

$$\min_{\mathbf{W}^T\left(\sum_{i=1}^{n}(x_i-\mu)(x_i-\mu)^T\right)\mathbf{W}=\mathbf{I}} -tr\left(\mathbf{W}^T\left(\sum_{k=1}^{c}n^k\left(\mu^k-\mu\right)\left(\mu^k-\mu\right)^T\right)\mathbf{W}\right)+\gamma\left\|\mathbf{W}\right\|_{2,1}$$

$$\Rightarrow \min_{\sum_{i=1}^{n}\mathbf{W}^T(x_i-\mu)(x_i-\mu)^T\mathbf{W}=\mathbf{I}} -tr\left(\sum_{k=1}^{c}n^k\mathbf{W}^T\left(\mu^k-\mu\right)\left(\mu^k-\mu\right)^T\mathbf{W}\right)+\gamma\left\|\mathbf{W}\right\|_{2,1}$$

$$\Rightarrow \min_{\sum_{i=1}^{n}\left\|(x_i-\mu)^T\mathbf{w}\right\|_2^2=\mathbf{I}} -\sum_{k=1}^{c}n^k\left\|\left(\mu^k-\mu\right)^T\mathbf{W}\right\|_2^2+\gamma\left\|\mathbf{W}\right\|_{2,1}. \quad (9)$$

As shown in (9), it is clear that the objective involves the squared l_2-norm terms. It is well known that squared l_2-norm is sensitive to the presence of outliers[31]. The estimates of the distances could be highly influenced by outlying data samples and outlying features. That is, the objective value is inappropriate on contaminated data sets, because the large squared error distances dominate the sum.

III. L21-NORM DISTANCE ROBUST FEATURE SELECTION

In this section, we will firstly develop a robust objective using $l_{21}-norm$ distances and solving the optimization problem. The objective is challenging to solve because it involves a series of $l_{21}-norm$ terms as well as it is non-smooth. Then, we introduce an iterative algorithm which is efficient to solve the problem. Also, the convergence will be proved next.

A. Learning the Robust Feature Selection

As proved above, the DFS is based on the squared l_2-norm disatence, although the $l_{21}-norm$ is introduced in the regularization term. It may lose the ability to choose the most discriminative features due to the existence of outlying features and outlying data samples. In our new method, the $l_{21}-norm$ distance is not only used in the regularization term, but also the objective term. As a result, the new method is more likely to provide better robustness and sparsity.

In this section, some notions and the definitions of the $l_{21}-norm$ are given at first, and then the objective function of our new method is presented. Ultimately, the algorithm and the related proof will be introduced.

For a matrix $X=[X_i^j]\in R^{m*n}$, we denote the ith row of X by X_i,which means the ith data point in the matrix. Similarly, the jth column of X is denoted by X^j and it indicates the jth feature of the data. Consequently, the value X_i^j presents the jth feature of the ith data point in matrix X.

For a matrix X, the traditional squaredl_2-norm distance of X is defined as

$$\left\|\mathbf{X}\right\|_2^2=\sum_{i=1}^{m}\mathbf{X}_i^2=\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\mathbf{X}_i^j\right)^2. \quad (10)$$

The$l_{21}-norm$ distance of matrix X is defined as

$$\left\|\mathbf{X}\right\|_{2,1}=\sum_{i=1}^{m}\left\|\mathbf{X}_i\right\|_2=\sum_{i=1}^{m}\sqrt{\sum_{j=1}^{n}\left(\mathbf{X}_i^j\right)^2}. \quad (11)$$

Similar to the traditional Linear Discriminant Analysis algorithm, the L21FS also needs to define the between-class scatter matrix and the within-class scatter matrix via $l_{21}-norm$ distance. Suppose the projective subspace is W, then the $l_{21}-norm$ distance between-class scatter value in the new subspace can be presented as follow:

$$\sum_{i=1}^{c}n_i\left\|\overline{\mathbf{X}_i}\mathbf{W}-\overline{\mathbf{X}}\right\|_2=\left\|\begin{array}{c}n_1\left(\overline{\mathbf{X}_1}-\overline{\mathbf{X}}\right)\mathbf{W}\\ \vdots\\ n_c\left(\overline{\mathbf{X}_c}-\overline{\mathbf{X}}\right)\mathbf{W}\end{array}\right\|_{2,1} \quad (12)$$

where $\bar{x}$ is the mean of the matrix X, c is the number of classes, $\bar{x}_i$ is the mean of the ith class data points, and n_i is the number of ith class data points. It is clear that the between-class data matrix X_b can be defined as

$$\mathbf{X}_b=\left[\begin{array}{c}n_1\left(\overline{\mathbf{X}_1}-\overline{\mathbf{X}}\right)\mathbf{W}\\ \vdots\\ n_c\left(\overline{\mathbf{X}_c}-\overline{\mathbf{X}}\right)\mathbf{W}\end{array}\right]. \quad (13)$$

Similarly, the projected $l_{21}-norm$ distance within-class scatter value can be defined as

$$\sum_{i=1}^{c}\sum_{j=1}^{n_i}\left\|\mathbf{X}_{ij}\mathbf{W}-\overline{\mathbf{X}}_i\mathbf{W}\right\|_2=\left\|\begin{array}{c}\left(\mathbf{X}_{11}-\overline{\mathbf{X}}_1\right)\mathbf{W}\\ \vdots\\ \left(\mathbf{X}_{1n_1}-\overline{\mathbf{X}}_1\right)\mathbf{W}\\ \vdots\\ \left(\mathbf{X}_{cn_c}-\overline{\mathbf{X}}_c\right)\mathbf{W}\end{array}\right\|_{2,1} \tag{14}$$

where the within-class data matrix

$$\mathbf{X}_w=\begin{bmatrix}\left(\mathbf{X}_{11}-\overline{\mathbf{X}}_1\right)\mathbf{W}\\ \vdots\\ \left(\mathbf{X}_{1n_1}-\overline{\mathbf{X}}_1\right)\mathbf{W}\\ \vdots\\ \left(\mathbf{X}_{cn_c}-\overline{\mathbf{X}}_c\right)\mathbf{W}\end{bmatrix}.$$

Recall the optimization criterion of LDA, it requires the objective minimize the within-class matrix value and maximize the between-class matrix value simultaneously. Using this idea, we can achieve the optimal projection matrix W through the following objective function:

$$\mathbf{W}^*=\arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}}\left\|\mathbf{X}_w\mathbf{W}\right\|_{2,1} \tag{15}$$
$$s.t.\ \left\|\mathbf{X}_b\mathbf{W}\right\|_{2,1}=cons.$$

In formula (15), the within-class scatter value will be fixed as a constant to facilitate the calculation. So far, we can get the optimal projection matrix of $l_{21}-norm$ distance by solving the minimum optimization problem. For W^*, each of its rows corresponds to a feature. If all elements of a row are zero, it means that the corresponding feature makes no contribution to the classification. To convert the $l_{21}-norm$ distance LDA into a feature selection method, we have to force more rows to be zero. Hence, it is necessary to introduce a $l_{21}-norm$ regularization:

$$\mathbf{W}^*=\arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}}\left\|\mathbf{X}_w\mathbf{W}\right\|_{2,1}+\gamma\left\|\mathbf{W}\right\|_{2,1} \tag{16}$$
$$s.t.\ \left\|\mathbf{X}_b\mathbf{W}\right\|_{2,1}=cons.$$

where $\gamma>0$ is a parameter which can regulate the row sparsity of the projection matrix. A larger γ means more rows are forced to be close to zero, vice versa. The objective (16) can be rewritten as

$$\mathbf{W}^*=\arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}}\frac{\left\|\mathbf{X}_w\mathbf{W}\right\|_{2,1}+\gamma\left\|\mathbf{W}\right\|_{2,1}}{\left\|\mathbf{X}_b\mathbf{W}\right\|_{2,1}}. \tag{17}$$

Although the motivation of the objective function is clear and simple, it is a non-smooth objective and difficult to be solved effectively. Hence, in the following section, we give an iterative algorithm to solve the minimization and maximization problem.

B. Algorithm for L21FS

Before deriving our new method, we will first introduce the following proposition.

Lemma 1: For any matrix X, when none of the rows is zero, we have the following equation[30]:

$$\left\|\mathbf{X}\right\|_{2,1}=trace\left(\mathbf{X}^T\mathbf{D}_\mathbf{X}\mathbf{X}\right) \tag{18}$$
$$s.t.\ \mathbf{D}_\mathbf{X}=diag\left(\frac{1}{\left\|\mathbf{X}_1\right\|},\ \cdots,\ \frac{1}{\left\|\mathbf{X}_m\right\|}\right).$$

According to Lemma 1, the between-class scatter value can be rewritten as

$$\sum_{i=1}^{c}n_i\left\|\overline{\mathbf{X}}_i\mathbf{W}-\overline{\mathbf{X}}\mathbf{W}\right\|_2=\left\|\mathbf{X}_b\mathbf{W}\right\|_{2,1}=tr\left(\mathbf{W}^T\mathbf{X}_b\mathbf{D}_b\mathbf{X}_b^T\mathbf{W}\right) \tag{19}$$

where D_b is a diagonal Matrix and defined as

$$\mathbf{D}_b=diag\left(\frac{1}{\left\|n_1\left(\overline{\mathbf{X}}_1-\overline{\mathbf{X}}\right)\mathbf{W}\right\|_2},\ \cdots,\ \frac{1}{\left\|n_c\left(\overline{\mathbf{X}}_c-\overline{\mathbf{X}}\right)\mathbf{W}\right\|_2}\right). \tag{20}$$

Then the between-class scatter matrix can be denoted by

$$\mathbf{S}_b=\mathbf{X}_b\mathbf{D}_b\mathbf{X}_b^T. \tag{21}$$

Similarly, the within-class scatter value can be rewritten as

$$\sum_{i=1}^{c}\sum_{j=1}^{n_i}\left\|\mathbf{X}_{ij}\mathbf{W}-\overline{\mathbf{X}}_i\mathbf{W}\right\|_2=\left\|\mathbf{X}_w\mathbf{W}\right\|_{2,1}=tr\left(\mathbf{W}^T\mathbf{X}_w\mathbf{D}_w\mathbf{X}_w^T\mathbf{W}\right) \tag{22}$$

and the diagonal matrix

$$\mathbf{D}_w=diag\left(\frac{1}{\left\|\left(\mathbf{X}_{11}-\overline{\mathbf{X}}_1\right)\mathbf{W}\right\|_2},\ \cdots\ \frac{1}{\left\|\left(\mathbf{X}_{1n_1}-\overline{\mathbf{X}}_1\right)\mathbf{W}\right\|_2},\ \cdots\ \frac{1}{\left\|\left(\mathbf{X}_{cn_1}-\overline{\mathbf{X}}_c\right)\mathbf{W}\right\|_2}\right). \tag{23}$$

The within-class matrix can be denoted by

$$\mathbf{S}_w=\mathbf{X}_w\mathbf{D}_w\mathbf{X}_w^T. \tag{24}$$

Also, according to the Lemma 1, we have

$$\left\|\mathbf{W}\right\|_{2,1}=trace\left(\mathbf{W}^T\mathbf{D}\mathbf{W}\right) \tag{25}$$

where

$$\mathbf{D}=diag\left(\frac{1}{\left\|\mathbf{W}_1\right\|_2},\cdots,\frac{1}{\left\|\mathbf{W}_i\right\|_2}\right). \tag{26}$$

Review the above formula of L21FS, it can be solved by the $l_{21}-norm$ distance formula:

$$\mathbf{W}^*=\arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}}\frac{tr\left(\mathbf{W}^T\mathbf{X}_w\mathbf{D}_w\mathbf{X}_w^T\mathbf{W}\right)+\gamma\times tr\left(\mathbf{W}^T\mathbf{D}\mathbf{W}\right)}{tr\left(\mathbf{W}^T\mathbf{X}_b\mathbf{D}_b\mathbf{X}_b^T\mathbf{W}\right)}$$

$$\mathbf{W}^*=\arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}}\frac{tr\left(\mathbf{W}^T\mathbf{S}_w\mathbf{W}\right)+\gamma\times tr\left(\mathbf{W}^T\mathbf{D}\mathbf{W}\right)}{tr\left(\mathbf{W}^T\mathbf{S}_b\mathbf{W}\right)}. \tag{27}$$

Hence, the objective function can be solved with the eigenvalue problem. The optimal projection matrix W^* is the eigenvectors corresponding to the smallest eigenvalues:

$$\left(\mathbf{S}_b\right)^{-1}\left(\mathbf{S}_w+\gamma\mathbf{D}\right)\mathbf{W}=\mathbf{W}\Lambda. \tag{28}$$

C. An Efficient Algorithm to solve L21FS

Note that both S_w, S_b and D depend on the projection matrix W and thus they are also unknown variables. We proposed an

iterative algorithm in this section to obtain the solution of formulation (16) and (17), and prove that the algorithm will monotonically decreases the objective value in the next section.

ALGORITHM I
AN ITERATIVE ALGORITHM TO SOLVE L21FS

Input: Data X, and the parameter γ.

1. Initialize the column-orthogonal projection matrix W, and parameter γ.
2. Compute the between-class data matrix X_b and the within-class data matrix X_w.

while not converge

3. Compute D_b, D_w and D.
4. Construct the scatter matrix S_b, D_w from D_b, D_w.
5. Solve the problem of (28) to obtain the eigenvectors corresponding to the smallest eigenvalues.
6. Update W with the obtained eigenvectors.

end while

Remark 1: Since the rank of S_b is equal to the number of categories minus one $(c-1)$, that is, S_b is rank deficient, then the process of inversion of S_b will inevitably have singularity problems. To deal with this problem, we add a small value ς to the main diagonal elements of S_b.

Remark 2: Note that, in real problem some rows of W would be zero, it leads some elements of D_b, D_w and D to be non-extent. Similarly, we replace the $\|w_i\|_2$ by $\sqrt{w_i^2+\varsigma}$.

D. Convergence analysis of the algorithm

In this subsection, we will show that this algorithm forces the objective value to be decremented each time until it converges. Firstly, the following lemma inspired by [19] will be introduced.

Lemma 2: For function $f(a,b)=a-\frac{a^2}{2c}+\kappa b-\kappa\frac{b^2}{2d}$, given any nonzero value $a,b,c,d\neq 0\in R^n$ and $\kappa\geq 0$,the following inequality holds:

$$f\left(a,b\right)\leq f\left(c,d\right). \tag{29}$$

Hence, for any nonzero vectors $v,u,\tilde{v},\tilde{u}$, we have:

$$v-\frac{v^2}{2\tilde{v}}+ku-k\frac{u^2}{2\tilde{u}}\leq\tilde{v}-\frac{\tilde{v}^2}{2\tilde{v}}+k\tilde{u}-k\frac{\tilde{u}^2}{2\tilde{u}}. \tag{30}$$

Theorem 1: In the case of a fixed value $tr(W^TS_bW)$, this algorithm will decrease the objective value of (16) in each iteration till it converges to the local optimal W.

Proof: First, we denote the updated w by $\widetilde{W}$. In each iteration, we have

$$\tilde{\mathbf{W}}=\arg\min_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} tr\left(\mathbf{W}^T\mathbf{X}_w^T\mathbf{D}_w\mathbf{X}_w\mathbf{W}\right)+\gamma\times tr\left(\mathbf{W}^T\mathbf{D}\mathbf{W}\right) \tag{31}$$

$$s.t.\ tr\left(\mathbf{W}^T\mathbf{X}_b^T\mathbf{D}_b\mathbf{X}_b\mathbf{W}\right)=cons,$$

which indicates that

$$tr\left(\tilde{\mathbf{W}}^T\mathbf{X}_w^T\mathbf{D}_w\mathbf{X}_w\tilde{\mathbf{W}}\right)+\gamma\times tr\left(\tilde{\mathbf{W}}^T\mathbf{D}\tilde{\mathbf{W}}\right) \tag{32}$$

$$\leq tr\left(\mathbf{W}^T\mathbf{X}_w^T\mathbf{D}_w\mathbf{X}_w\mathbf{W}\right)+\gamma\times tr\left(\mathbf{W}^T\mathbf{D}\mathbf{W}\right).$$

For convenience, we denote the ith row of matrix W as w_i. That is to say,

$$\sum_i\frac{\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|^2}{2\left\|\mathbf{X}_w\mathbf{W}_i\right\|}+\gamma\sum_i\frac{\left\|\tilde{\mathbf{W}}_i\right\|^2}{2\left\|\mathbf{W}_i\right\|}\leq\sum_i\frac{\left\|\mathbf{X}_w\mathbf{W}_i\right\|^2}{2\left\|\mathbf{X}_w\mathbf{W}_i\right\|}+\gamma\sum_i\frac{\left\|\mathbf{W}_i\right\|^2}{2\left\|\mathbf{W}_i\right\|}$$

$$\Rightarrow\sum_i\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|-\left(\sum_i\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|-\sum_i\frac{\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|^2}{2\left\|\mathbf{X}_w\mathbf{W}_i\right\|}\right)$$

$$+\gamma\left[\sum_i\left\|\tilde{W}_i\right\|-\left(\sum_i\left\|\tilde{W}_i\right\|-\sum_i\frac{\left\|\tilde{\mathbf{W}}_i\right\|^2}{2\left\|\mathbf{W}_i\right\|}\right)\right]$$

$$\leq\sum_i\left\|\mathbf{X}_w\mathbf{W}_i\right\|-\left(\sum_i\left\|\mathbf{X}_w\mathbf{W}_i\right\|-\sum_i\frac{\left\|\mathbf{X}_w\mathbf{W}_i\right\|^2}{2\left\|\mathbf{X}_w\mathbf{W}_i\right\|}\right)$$

$$+\gamma\left[\sum_i\left\|W_i\right\|-\left(\sum_i\left\|W_i\right\|-\sum_i\frac{\left\|\mathbf{W}_i\right\|^2}{2\left\|\mathbf{W}_i\right\|}\right)\right]. \tag{33}$$

According to (30), we have that

$$\sum_i\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|-\sum_i\frac{\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|^2}{2\left\|\mathbf{X}_w\mathbf{W}_i\right\|}+\gamma\sum_i\left\|\tilde{\mathbf{W}}_i\right\|-\gamma\sum_i\frac{\left\|\tilde{\mathbf{W}}_i\right\|^2}{2\left\|\mathbf{W}_i\right\|} \tag{34}$$

$$\leq\sum_i\left\|\mathbf{X}_w\mathbf{W}_i\right\|-\sum_i\frac{\left\|\mathbf{X}_w\mathbf{W}_i\right\|^2}{2\left\|\mathbf{X}_w\mathbf{W}_i\right\|}+\gamma\sum_i\left\|\mathbf{W}_i\right\|-\gamma\sum_i\frac{\left\|\mathbf{W}_i\right\|^2}{2\left\|\mathbf{W}_i\right\|}.$$

Combing (33) and (34), we obtain

$$\sum_i\left\|\mathbf{X}_w\tilde{\mathbf{W}}_i\right\|+\gamma\sum_i\left\|\tilde{\mathbf{W}}_i\right\|\leq\sum_i\left\|\mathbf{X}_w\mathbf{W}_i\right\|+\gamma\sum_i\left\|\mathbf{W}_i\right\|. \tag{35}$$

That is to say,

$$\left\|\mathbf{X}_w\tilde{\mathbf{W}}\right\|_{2,1}+\gamma\left\|\tilde{\mathbf{W}}\right\|_{2,1}\leq\left\|\mathbf{X}_w\mathbf{W}\right\|_{2,1}+\gamma\left\|\mathbf{W}\right\|_{2,1}. \tag{36}$$

Thus the algorithm will monotonically decrease the objective value of the problem (16) in each iteration with the constraint of $tr(W^TS_bW)=cons$. Note that the objective function (16) must be greater than 0, which means that it has the lower bound. Therefore, the algorithm will monotonically decrease the objective value until it converges to the local optimal W of the problem (16).

E. Computational Complexity Analysis

In the process of optimizing the function of L21FS, the most time consuming operation is to solve the eigen-problem $(S_b)^{-1}(S_w+\gamma D)W=W\Lambda$ in step 6. The time complexity of this operation is $O(n^3)$ approximately. Since the algorithm is an iterative algorithm, then the whole computational complexity is related to the number of iterations of the algorithm. Empirically, the experimental results demonstrate that the algorithm only needs several iterations to converge. Hence, the proposed method scales well in practice.

F. Evaluation Principle

Once we achieve the optimal projection matrix W^*, the next thing is to determine the evaluation principle of the feature importance. In this paper, we rank features according to the

Euclidean Metric of each rows in descending order. That is to say, if the value of $\|W_i\|_2$ is bigger, then the corresponding feature is more important. With this principle, we can achieve the top ranked features as we desired.

IV. Experiments

In this section, we conducted a number of experiments to evaluate the performance of our new method. All codes are written in MATLAB_R2014b. The experimental environment: 2.7 GHz Intel Core i5 CPU, 8 GB 1867 MHz DDR3 memory. We employed the LIBSVM algorithm [32] to classify the data points. For accuracy, the testing accuracies for our method are computed using the traditional 10-fold cross validation [33]. The parameters of the compared methods are selected on the tuning set random 10% of the training data by 10-fold cross validation.

Firstly, we conduct two toy experiments to show the ability of our new algorithm to find the most discriminative features. Then we compare our L21FS method against several related state-of-the-art feature selection methods. Following this, we evaluate the performance of L21FS with varying the parameter λ. In order to further investigate the classification accuracies between our new method and other methods, we also employ the paired t-tests method. Then, we experiment on the data sets with noisy data and noisy features respectively. In the end, we study the convergence curves of our proposed method.

A. Data Description

In our experiments, several widely used benchmark data sets are employed, including the ORL[1], USPS[1], MADELON[1], LUNG_DISCRETE[1], ISOLET5[2], ISOLET[1], COIL20[1] and COLON[1]. All the data sets are introduced as follows,

1) ORL contains a set of face images taken between April 1992 and April 1994 at the lab, the size of each image is 32×32. There are ten different images of each of 40 distinct people.
2) USPS is a popular subset contains 9298 16x16 handwritten digit images in total, which is then split into 7291 training images and 2007 test images.
3) MADELON is an artificial dataset, which was part of the NIPS 2003 feature selection challenge. This is a two-class classification problem with continuous input variables. The difficulty is that the problem is multivariate and highly non-linear.
4) ISOLET5 and ISOLET contain 150 subjects who 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.
5) COIL20 contains 20 objects. The images of each objects were taken 5 degrees apart as the object is rotated on a turntable and each object has 72 images. The size of each image is 32x 32 pixels, with 256 grey levels per pixel. Thus, each image is represented by a 1024-dimensional vector.
6) COLON contains 62 samples collected from colon-cancer patients. Among them, 40 tumor biopsies are from tumor and 22 normal biopsies are from healthy parts of the colons of the same patients. Two thousand out of around 6500 genes were selected based on the confidence in the measured expression levels.

[1]http://featureselection.asu.edu/datasets.php.
[2]https://archive.ics.uci.edu/ml/datasets/isolet.
[3]https://archive.ics.uci.edu/ml/datasets/iris.

B. Toy Example ORL

In order to visually show the ability of our new method to be able to select the most distinguishing features, we present a toy experiment on ORL data set, which collects face pictures of 40 persons. Here we randomly select two people photo data as training data. To draw the pictures, the top ranked $\{32,64,128,256,512,640,768,896,1024\}$ features are selected. For illustration, the unselected features are presented by white points and the selected features are presented by the original values. In the Fig 1, the first line is repainted one person and the last line is another.

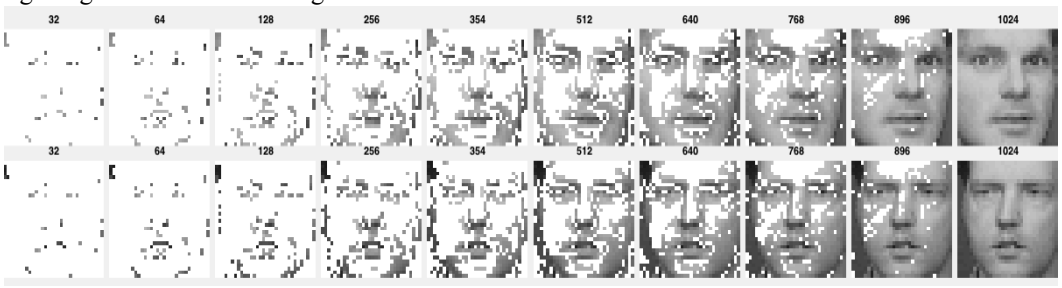


Fig. 1. Toy examples

From Fig 1, we can find that only with 64 features, a picture is clear enough to recognize a person. Particularly, the 64 features clearly show the eyes, nose and mouth, which are the most discriminative parts of a face. Also, we note that the features do not gather together or distribute widely, they just show the crucial parts to classify different people. This has strongly demonstrated the ability of L21FS to select the most robust and discriminative features, which is consistent with our vision.

C. Toy Example Iris

The Iris[3] dataset is a popular dataset on the UCI ML library with a total of 3 categories of 50 samples for each category. Each sample contains four features, representing sepal length, sepal width, petal length and petal width. Due to the simplicity

and popularity of this data set, we experiment on it to express the effect of our proposed method intuitively.

In this experiment, we selected two features from the Iris dataset and then plotted each point in a two-dimensional coordinate system. We have traced all possible combinations of these four features and plotted them. Besides, the samples presented by the selected features we achieved using L21FS are also plotted. All the pictures are showed in Fig 2.

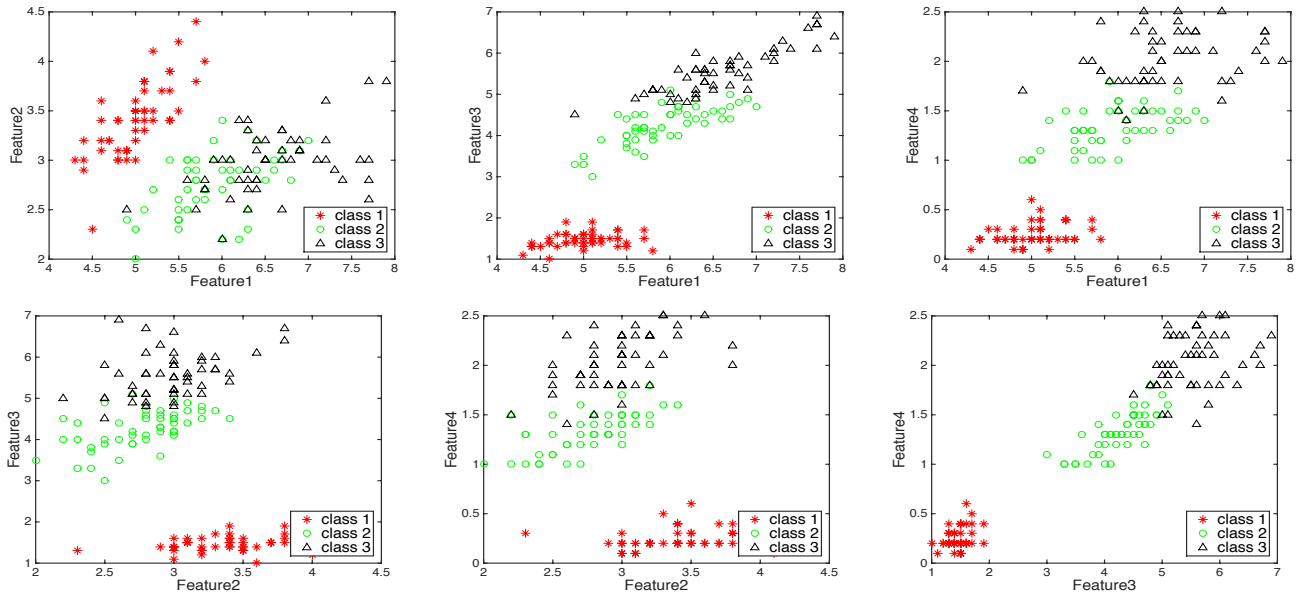


Fig.2. (a) Six possibilities for 2-D Iris data presented by two features.

As seen from Fig 2, L21FS picked the two features that most visually distinguished the three categories in these six possibilities. When we take a close look at Fig 2 (b), we can find that the points of the same class are grouped together, and points between different classes are far apart. This phenomenon is consistent with the idea of traditional LDA. Hence, the toy experiment on the Iris dataset is a good indicator of the ability of L21FS to select the most distinguishing features.

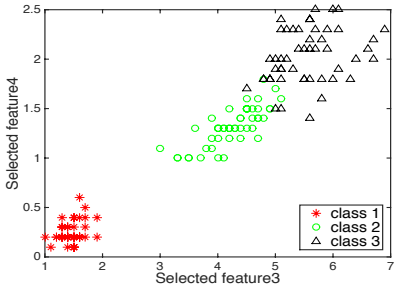


Fig. 2. (b) 2-D Iris data presented by the selected features using L21FS.

D. Numerical Testing and Comparisons

In order to demonstrate the performance of our new method, we report the results on public data sets and compare our approach against with four widely used and latest feature selection methods:

Discriminative Feature Selection (DFS), which improves the LDFS by $l_{2,p} - norm$ regularization to avoid the trivial solutions, and has the ability of selecting the most discriminative features and removing the redundant ones simultaneously.

Laplace Score (LS) [34-36], which evaluates the importance of each feature by its contribution to locality preserving and selects the top ranked ones.

Multi-Cluster Feature Selection (MCFS) [37], which selects the features such that the multi-cluster structure of the data can be preserved. Different from the traditional ranking methods, MCFS selected the best features with the help of manifold learning and L1-regularized models.

Unsupervised Maximum Margin (UMM) [38], which combines the feature selection and K-means cluster method to select the most discriminative subspace.

In order to illuminate the effect of our new method, we need the following several metrics:

Accuracy: We adopt the 10-fold cross-validation strategy to evaluate the performance of each method. During each experiment, the data set would be split into ten equal sized subsets for training and testing. The average accuracy on the 10 classification tasks would represent the effect of corresponding method.

Time: The average running time represents the computational cost.

Variance: The smaller the variance indicates that the algorithm has better robustness and is less affected by the data.

Statistical significance: We perform the paired t-tests comparing L1FS with other methods. The p-value of t-tests stands for the probability of a great difference between two classification accuracy values. The smaller the p-value, the less likely that the observed difference between the two methods. A typical threshold for p-value is 0.05 [39]. For example, if p-value is smaller than 0.05, it means that there is a great difference between the two methods, and vice versa.

For DFS and L21FS, the projection matrix dimension is set as $c-1$, just like the traditional LDA does. For all the parameters of the five algorithms, we obtain them by 10-fold cross validation. We employ the LIBSVM to classify the samples presented by the selected features, using the 5-fold cross validation. The average accuracies of each algorithm are reported as the final accuracies in Table 2 and Table 3. The best performance is marked in bold.

Table 2. Accuracy of top 20 features. (average ± STD, time: s, p-value)

TABLE 2
ACCURACY OF TOP 20 FEATURES. (AVERAGE ± STD, TIME: S, P-VALUE)

	UMM	MCFS	LS	DFS	L21FS
USPS (2007x256, class:10)	73.94±2.52	87.84±1.58	53.31±0.98	**89.98±1.74**	89.63±1.67
	2.9441	0.1151	0.1284	**0.3622**	0.4324
	6.5402e-6	0.1574	2.8636e-10	**0.7805**	—
MADELON (2000x500,class:2)	61.00±1.65	60.25±1.30	61.10±1.57	60.60±0.93	**61.30±1.65**
	4.7316	0.0341	0.1872	1.9471	**2.3182**
	0.8038	0.3479	0.8654	0.4817	—
LUNG_DISCRETE (73x325,class:7)	67.04±8.1838	76.57±6.1677	57.52±8.78	71.23±2.43	**87.52±5.50**
	0.2948	0.0195	0.0027	0.6794	**0.8650**
	0.0032	0.0293	4.1081e-4	6.4027e-4	—
ISOLET5 (1559x617,class:26)	38.99±6.41	73.76±2.85	43.23±4.72	71.32±4.00	**76.77±2.06**
	4.6121	0.5380	0.1419	3.5905	**3.7723**
	3.5866e-6	0.1257	1.1493e-6	0.0419	—
ISOLET (1559x617,class:26)	33.07±3.22	73.58±3.11	54.93±4.92	68.33±5.89	**79.55±2.86**
	4.5595	0.5644	0.1404	3.6048	**4.1939**
	2.2656e-8	0.0227	2.4978e-5	0.0091	—
COIL20 (1440x1024,class:20)	67.08±1.93	87.56±3.70	61.66±4.95	**94.93±1.72**	91.66±2.48
	11.2796	0.7159	0.1890	**15.2525**	13.6754
	2.8113e-7	0.1034	4.6702e-6	**0.00629**	—
COLON (62x2000,class:2)	70.76±15.60	80.76±10.81	60.89±15.90	64.23±13.00	**85.38±6.32**
	47.2229	0.0616	0.0067	34.4736	**43.1703**
	0.1207	0.4822	0.0211	0.0191	—

TABLE 3
ACCURACY OF TOP 40 FEATURES. (AVERAGE ± STD, TIME: S, P-VALUE)

	UMM	MCFS	LS	DFS	L21FS
USPS (2007x256, class:10)	83.10±1.10	90.63±1.77	66.11±4.81	**91.62±1.23**	91.18±1.18
	2.9376	0.3461	0.1557	**0.3892**	0.7723
	8.7859e-6	0.6210	7.8923e-6	**0.6150**	—
MADELON (2000x500,class:2)	60.80±2.01	58.65±2.32	60.35±1.72	59.95±1.74	**60.85±1.55**
	4.8755	0.0590	0.1843	1.6079	**2.3469**
	0.9696	0.1545	0.6785	0.4628	—
LUNG_DISCRETE (73x325,class:7)	71.14±8.41	79.23±8.18	64.28±9.20	71.14±5.48	**84.85±3.16**
	0.3084	0.0482	0.0034	0.6491	**0.9327**
	0.0158	0.2362	0.0029	0.0025	—
ISOLET5 (1559x617,class:26)	49.90±4.00	86.14±1.36	63.05±2.94	82.36±3.11	**86.40±2.34**
	4.6838	1.1678	0.1440	3.6057	**5.3912**
	2.6778e-7	0.8540	1.6578e-6	0.0718	—
ISOLET (1559x617,class:26)	45.25±2.06	86.02±1.81	63.58±2.54	80.89±3.25	**89.03±3.11**
	4.7121	1.2450	0.1366	3.6701	**4.4487**
	1.1557e-8	0.1330	1.4194e-6	0.0068	—
COIL20 (1440x1024,class:20)	72.63±2.62	95.55±1.21	68.61±0.47	**97.22±1.07**	96.73±0.99
	11.4341	1.5625	0.1853	**15.4507**	16.8142
	1.3667e-7	0.1706	2.4107e-11	**0.5260**	—
COLON (62x2000,class:2)	72.30±18.73	83.97±11.53	64.23±15.89	64.23±13.00	**85.38±6.32**
	49.3434	0.0670	0.0066	40.8704	**46.3781**
	0.2225	0.8356	0.0385	0.0191	—

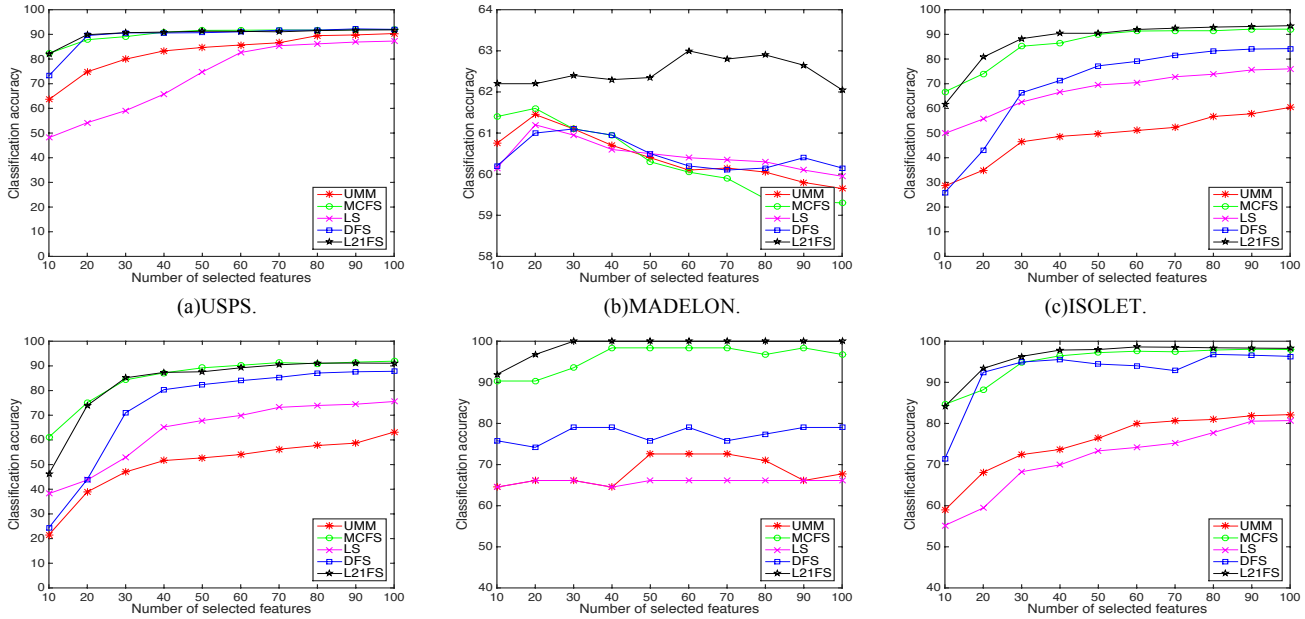
Table 2 and Table 3 show the details of the classification accuracy using the top 20 and 40 features on the seven datasets respectively. As shown in the two tables, the L21FS performed best compared with the other four feature selection methods. On these seven data sets, L1FS performed best on five data sets, DFS in two data sets on the best. One point should be noticed here. We find four cases wherein DFS has better average accuracies comparing the proposed algorithms, whereas almost all the corresponding t-test p-values are less than 0.05. It indicates that in these data sets, there is no essential difference between the performance of the two methods although the numerical shows a slight difference. In contrast, the p-values are always smaller than 0.05 in most cases. The p-values demonstrate that the algorithm we proposed is obviously different from the other four algorithms in statistical significance. Besides, the standard deviation of L21FS is

always smaller than the compared methods, leading us to conclude that L21FS are more stable than other methods and have better robustness.

When concentrating on the computational consuming shown in Table 1 and Table 2, we find that MCFS and LS consume much less time than the other three algorithms. This can be explained in terms of computational complexity. For MCFS, its time complexity is about $O(n^2m)$. For LS, the most time consuming step is to compute the Rayleigh Quotient and the corresponding time complexity is $O(n^3)$. Considering $n \ll m$ and the other three algorithms are all iterative methods, the difference in time consuming can be well explained.

Also, we compare our algorithm with the other four algorithms with different number of features. The classification accuracy versus the variations of the selected features are shown in Fig 3.

From Fig 3, we can see that L21FS usually achieve higher classification accuracy in a low dimensional subspace compared with other feature selection methods. This phenomenon is consistent on the six datasets and more salient in COLON, COIL20 and MADELON datasets. The result shown in Fig 3 visually indicates that the L21FS do have a better ability of feature selection than previous algorithms.



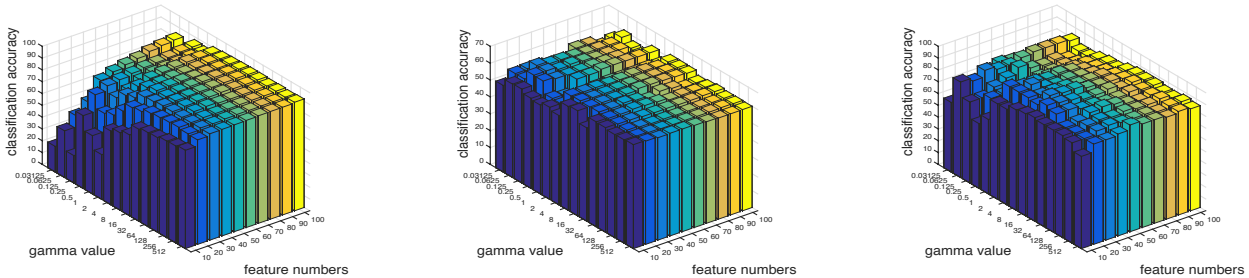
(a)USPS. (b)MADELON. (c)ISOLET.

(d)ISOLET5. (e)COLON. (f)COIL20.

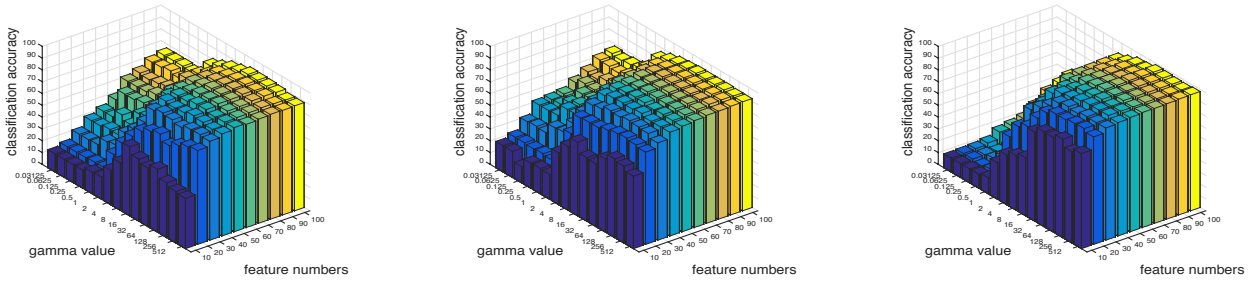
Fig 3. Classification accuracy versus the variations of the selected features

E. Impact of γ on The Performance of L21FS

In this proposed new method, there is only one parameter γ, which can balance the sparsity and the convexity of the formulation of L21FS. The bigger the values of γ is, the sparser the L21FS is, that is to say, the more rows of the projection matrix W are forced to be zero. In this subsection, we focus mainly on the impact of γ on the performance of our new method. We vary the γ value from the minimum value of 2^{-7} to the maximum value of 2^7, each interval the γ value is doubled. Without losing the generality, we select the top [10,20,30,40,50,60,70,80,90,100] features in all our experiments on datasets COLON, ISOLET5 and COIL20. The performance variance w.r.t. γ and the number of selected features is showed in Fig 4.



(a)USPS (b)MADELON (c)LUNG_DISCRETE



(d)ISOLET5 (e)ISOLET (f)COIL20

Fig. 4. Performance variations of L21FS w.r.t. different value of the papramter γ.

As shown in Fig 4, L21FS has similar performance variance trends w.r.t the regularization parameter on each data sets, but has the different optimal parameter γ. Overall, with the same number of features, the greater the value of γ, the higher the accuracy will be. It is noteworthy that, when the γ value reaches 16 or 32, the accuracy has been able to reach a high level, then the increasing of γ has little effect on the accuracy. Besides, when the number of selected features is small, the performance of our method is more sensitive to the γ value. That is to say, the performance variance created by the regularization parameter is bound up with the numbers of the selected features.

F. Accuracy on Data with Outlier Samples

Since our proposed method is a robust feature selection method, then we must experiment with the noisy data. To emulate the outlier data samples, we corrupt the input data set $X = [X_1; \cdots; X_m] \in R^{m \times n}$ by a noise matrix $\tilde{X} \in R^{m \times n}$ whose elements are i.i.d standard Gaussian variables. Then we conduct the same experiments on the corrupted data $X + \theta\tilde{X}$ as the original data X, where $\theta = nf\frac{\|X\|_F}{\|\tilde{X}\|_F}$ and nf is a given noise factor . In this subsection, we set $nf = 0.1$ in all our experiments. We compare our method against the 4 competing methods using the same experimental settings as before and report the results in the Table 4 and Table 5.

TABLE 4
ACCURACY OF TOP 20 FEATURES WITH NOISE (AVERAGE ± STD, TIME: S, P-VALUE)

	UMM	MCFS	LS	DFS	L21FS
USPS (2007x256, class:10)	73.64±2.33 3.0750 5.1935e-6	87.94±0.89 0.1313 0.2255	53.71±1.55 0.1311 1.5484e-9	**89.33±1.20** **0.1498** **0.9284**	89.2±1.75 0.1754 —
MADELON (2000x500,class:2)	61.05±1.60 5.1963 0.7870	60.20±1.20 0.0373 0.3222	60.95±1.52 0.1935 0.7239	56.90±1.67 0.5634 0.0078	**61.40±1.92** **0.9652** —
LUNG_DISCRETE (73x325,class:7)	67.14±7.66 0.3043 0.0022	76.57±6.16 0.0241 0.0360	57.52±12.17 0.0049 0.0022	67.04±8.18 0.2602 0.0031	**84.95±2.51** **0.8871** —
ISOLET5 (1559x617,class:26)	38.99±4.58 4.7873 4.3673e-7	71.13±3.22 0.5766 0.0834	42.07±3.73 0.1471 2.1851e-7	53.36±1.76 1.2686 8.4249e-8	**74.66±1.52** **2.3271** —
ISOLET (1559x617,class:26)	31.02±2.58 4.7768 1.6676e-8	73.91±5.82 0.6071 0.2019	54.55±4.78 0.1456 3.6108e-5	50.32±3.59 1.4804 3.0244e-6	**78.58±3.37** **2.1248** —
COIL20 (1440x1024,class:20)	67.63±1.85 11.6559 1.0762e-8	89.86±2.94 0.7300 0.1455	58.19±3.17 0.1829 3.2531e-8	89.93±0.93 5.7540 0.0066	**92.08±2.38** **5.7619** —
COLON (62x2000,class:2)	70.64±16.42 46.4025 0.4789	73.97±13.59 0.1317 0.6730	61.02±13.74 0.0067 0.0734	65.64±18.06 37.8078 0.2672	**77.43±8.06** **40.3341** —

TABLE 5
ACCURACY OF TOP 40 FEATURES WITH NOISE (AVERAGE ± STD, TIME: S, P-VALUE)

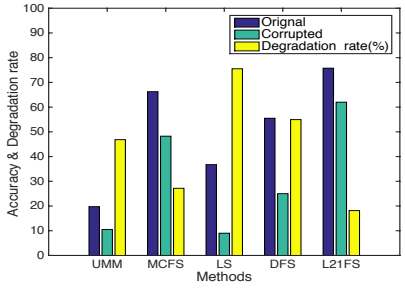
	UMM	MCFS	LS	DFS	L21FS
USPS (2007x256, class:10)	83.20±1.82 2.9615 9.8808e-5	89.93±1.71 0.3138 0.4292	70.20±3.10 0.1310 1.5564e-6	**92.17±1.41** **0.1548** **0.1571**	90.78±1.09 0.2387 —
MADELON (2000x500,class:2)	60.60±1.98 4.8088 0.9446	59.00±2.23 0.0567 0.2857	60.50±1.70 0.1826 0.8813	58.70±1.65 0.5200 0.1573	**60.70±1.95** **0.5276** **—**
LUNG_DISCRETE (73x325,class:7)	69.80±8.49 0.3220 0.0293	76.57±7.47 0.0436 0.1848	64.28±9.20 0.0024 0.0078	72.47±9.15 0.2646 0.0778	**83.42±5.78** **0.4309** **—**
ISOLET5 (1559x617,class:26)	50.61±4.74 4.7239 8.7922e-7	84.46±4.58 1.2656 0.5914	62.15±1.16 0.1414 6.1962e-8	65.29±4.17 1.5309 2.3321e-5	**85.8±2.21** **2.0919** **—**
ISOLET (1559x617,class:26)	45.00±2.17 4.7068 2.5932e-9	87.37±1.67 1.1739 0.4005	64.03±2.23 0.1444 2.6152e-7	70.25±5.06 1.2809 1.6017e-4	**88.58±2.17** **2.3248** **—**
COIL20 (1440x1024,class:20)	72.98+2.69 11.4180 2.9313e-7	95.69±0.71 1.4816 0.3171	67.22±3.23 0.1872 1.7285e-7	93.05±1.45 5.7293 0.0087	**96.52±1.38** **7.7865** **—**
COLON (62x2000,class:2)	72.30±2.69 11.4180 0.3634	78.84±10.27 0.0787 0.6272	64.23±15.89 0.0064 0.0802	72.30±15.75 39.5869 0.2996	**82.17±8.29** **41.0819** **—**

From Table 4 and Table 5, we have the following interesting observations. First, our method performed the best on the most cases of the seven experimental data sets, which demonstrate that the L21FS method is more robust than the other compared methods and is more possible to learn the most discriminative features with noisy data. Second, although our algorithm performs only slightly better than other algorithms on the original data sets, our algorithm has the lowest classification accuracy decline in the case of noisy data. Also, when taking a close look at the standard variation value, the standard variation of L21FS is always extremely smaller than the standard variations of the competing methods. This also fully demonstrates the robustness of our algorithm.

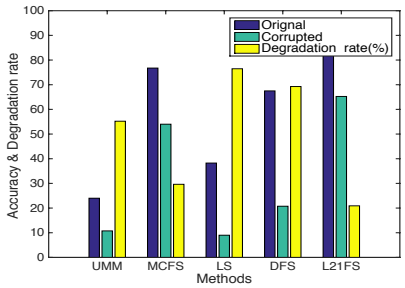
G. Accuracy on Data with Outlier Features

As introduced before, our method replaces the distance metric method not only in the regularization term, but also the learning objective. The $l21-norm$ distance feature selection method is robust against both outliers and the noisy features. Therefore, in this subsection, we conduct experiments to test the feature robustness of our proposed method on face image data set (ORL). In order to evaluate the feature robustness, a black square of size 8×8 is randomly placed onto the image to emulate the corrupted features. The occluded images are showed in the Fig 5. We selected the top 20 and top 40 features respectively for classification and the accuracy and the degradation rate are reported in the Fig 6.

Fig. 5. Six randomly selected original images VS the corresponding corrupted images.



(a)with top 20 features.



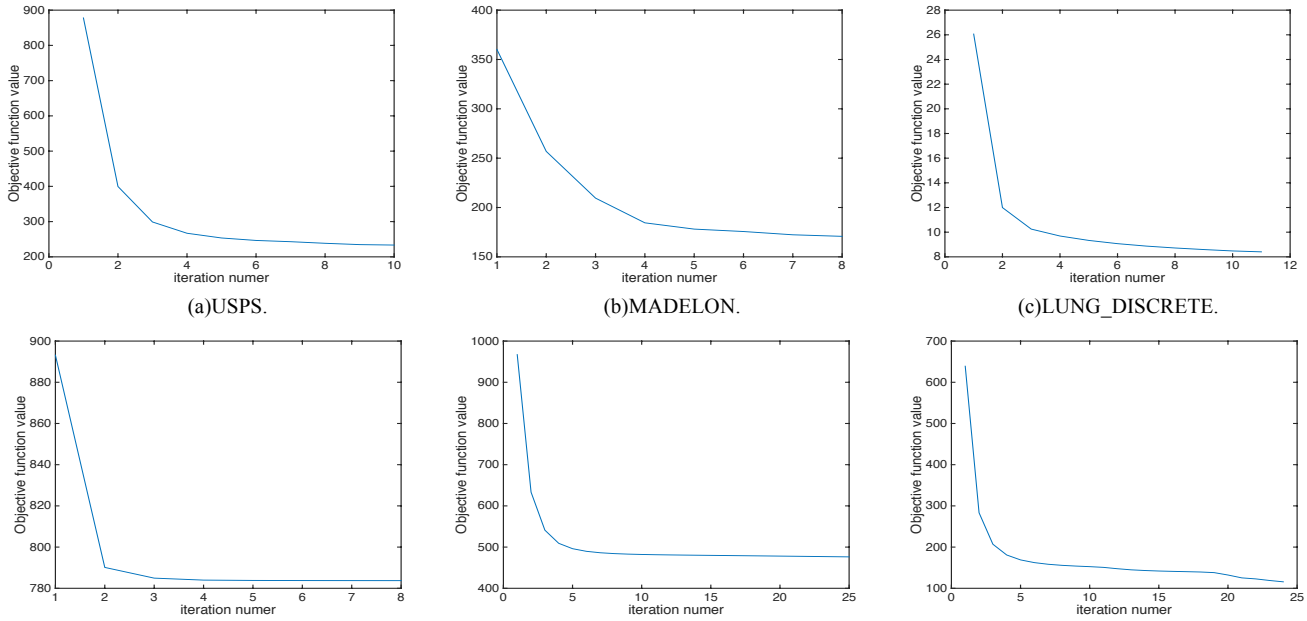
(b) with top 40 features.

Fig. 6. Recognition rates and the decline rates with top 20 and 40 features respectively.

As indicated in the Fig 6, our method outperformed the competing methods on both original images and the corrupted images. Moreover, when focusing on the degradation, we notice that the performance degradation of our proposed methods is very small, which provide more concrete evidence to support the robustness of L21FS.

H. Convergence Study

Finally, we evaluate the computational efficiency of our proposed method by presenting the objective convergence behavior curves on the experimental data sets. As analyzed in the previous section, the objective function value will converge to a local optimum. Hence, the number of iterations is one of the most important part of the efficiency of our proposed method and it determines how fast our algorithm converges. The objective function value convergence curves on the six data sets are plotted in Fig 7.



(a)USPS. (b)MADELON. (c)LUNG_DISCRETE.

(d)ISOLET5. (e)ISOLET. (f)COIL20.

Fig 7. Objective function value of L21FS vs. number of iterations.

Naturally, the objective function values of our proposed method on all of the data sets keep to decrease along with the iteration processes show shown in Fig 7, which is in accordance with our earlier theoretical analysis perfectly. Moreover, we can find that the algorithm typically converges to a local optimal within about 7 iterations. This small number of iterations ensures the efficiency and feasibility of our proposed algorithm. Hence, our proposed L21FS method scales well in practice due to the fast convergence speed.

V. CONCLUSIONS

In this paper, we proposed a new feature selection method which combines the traditional feature transformation method LDA and the $l_{21} - norm$ distance matric. Different from the previous feature selection methods with $l_{21} - norm$ regularization term, we also employed the $l_{21} - norm$ distance in our learning function. The novel objective function formulated a non-smooth non-convex optimal problem. In order to maximizing the $l_{21} - norm$ distance within-class scatter value and minimizing the $l_{21} - norm$ distance between-class scatter value simultaneously, we introduced an efficient iterative algorithm. The rigorous theoretical convergence proof and the computational consuming analysis indicate that l_{21} FS is efficient and fast to converge. The extensive experimental results show that our proposed method outperforms the related state-to-art methods. Moreover, experiments on various types of data sets illustrate that our proposed method is robust to both noise data and noise features.

REFERENCES

[1] Guyon, et al., An introduction to variable and feature selection. Journal of Machine Learning Research, 2003. 3(6): p. 1157-1182.

[2] Kwak, N. and C.H. Choi, Input feature selection for classification problems. IEEE Transactions on Neural Networks, 2002. 13(1): p. 143.

[3] Liu, et al., Feature Selection for Knowledge Discovery and Data Mining. Springer International, 1998(4): p. xviii.

[4] Fukunaga, K. and D.R. Olsen, An Algorithm for Finding Intrinsic Dimensionality of Data. IEEE Transactions on Computers, 2006. C-20(2): p. 176-183.

[5] Levina, E. and P.J. Bickel, Maximum Likelihood Estimation of Intrinsic Dimension. 2004. 17.

[6] Bazaraa, M.S., H.D. Sherali, and C.M. Shetty, Nonlinear Programming: Theory and Algorithms. Journal of the Operational Research Society, 1979. 30(11): p. 1025-1025.

[7] Duda, R.O. and P.E. Hart, Pattern recognition and scene analysis. 1973.

[8] Yao, C., et al., Local Regression and Global Information-Embedded Dimension Reduction. IEEE Transactions on Neural Networks & Learning Systems, 2018. PP(99): p. 1-12.

[9] Shi, X., et al., A Framework of Joint Graph Embedding and Sparse Regression for Dimensionality Reduction. IEEE Trans Image Process, 2015. 24(4): p. 1341-55.

[10] Dash, M. and H. Liu, Feature Selection for Classification. 1997: IOS Press. 131-156.

[11] Li, X.B., J.Y. Li, and R.H. Wang, Dimensionality reduction using MCE-optimized LDA transformation. 2016. 1: p. I - 137-40.
[12] Guo, B., et al. Semi-Supervised Multi-label Dimensionality Reduction. in IEEE International Conference on Data Mining. 2017.
[13] Lin, Y., et al., Multi-label feature selection based on max-dependency and min-redundancy. Neurocomputing, 2015. 168(C): p. 92-103.
[14] Wang, H., et al., Fisher Discriminant Analysis With L1-Norm. IEEE Transactions on Cybernetics, 2013. 44(6): p. 828-842.
[15] Welling, M., Fisher Linear Discriminant Analysis. Department of Computer Science, 2009. 16(94): p. 237-280.
[16] Yang, J., J.Y. Yang, and H. Ye, Theory of Fisher Linear Discriminant Analysis and Its Application. 2003. 29(4): p. 481-493.
[17] Yu, Y., et al., Semi-supervised Multi-label Linear Discriminant Analysis. 2017.
[18] Hong, T., et al., Effective Discriminative Feature Selection With Nontrivial Solution. IEEE Transactions on Neural Networks & Learning Systems, 2016. 27(4): p. 796-808.
[19] Lai, Z., et al., Rotational Invariant Dimensionality Reduction Algorithms. IEEE Transactions on Cybernetics, 2017. 47(11): p. 3733.
[20] Rodriguez-Lujan, I., C. Santa Cruz, and R. Huerta, On the equivalence of Kernel Fisher discriminant analysis and Kernel Quadratic Programming Feature Selection. Pattern Recognition Letters, 2011. 32(11): p. 1567-1571.
[21] Sugiyama, M., Dimensionality Reduction of Multimodal Labeled Data by Local Fisher Discriminant Analysis. 2007: JMLR.org. 1027-1061.
[22] Wang, S., et al. A Feature Selection Method Based on Fisher's Discriminant Ratio for Text Sentiment Classification. in International Conference on Web Information Systems and Mining. 2009.
[23] Bishop, C.M., et al., Neural Network for Pattern Recognition. 1995.
[24] Masaeli, M., G. Fung, and J.G. Dy. From Transformation-Based Dimensionality Reduction to Feature Selection. in International Conference on Machine Learning. 2010.
[25] Wang, L., X. Shen, and Y.F. Zheng, On L_1-Norm Multi-class Support Vector Machines. Publications of the American Statistical Association, 2006. 102(478): p. 583-594.
[26] Yang, M.S., W.L. Hung, and T.I. Chung. Alternative Fuzzy Clustering Algorithms with L 1-Norm and Covariance Matrix. in International Conference on Advanced Concepts for Intelligent Vision Systems. 2006.
[27] Du, L., et al., Robust Multiple Kernel K-means Using L21-Norm. 2015.
[28] Kong, D., H. Huang, and H. Huang. Robust nonnegative matrix factorization using L21-norm. in ACM International Conference on Information and Knowledge Management. 2011.
[29] Lu, Z., et al. Robust Face Recognition Based l21-Norm Sparse Representation. in International Conference on Digital Home. 2014.
[30] Nie, F. and H. Huang, Non-Greedy L21-Norm Maximization for Principal Component Analysis. 2016.
[31] Wang, H., F. Nie, and H. Huang, Learning Robust Locality Preserving Projection via p-Order Minimization. 2015.
[32] Chang, C.C. and C.J. Lin, LIBSVM: A library for support vector machines. 2012. 2(3): p. 1-27.
[33] Kohavi, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. in International Joint Conference on Artificial Intelligence. 1995.
[34] Benabdeslem, K. and M. Hindawi. Constrained Laplacian Score for Semi-supervised Feature Selection. in Machine Learning and Knowledge Discovery in Databases - European Conference, Ecml Pkdd 2011, Athens, Greece, September 5-9, 2011. Proceedings. 2011.
[35] He, X., D. Cai, and P. Niyogi. Laplacian Score for Feature Selection. in International Conference on Neural Information Processing Systems. 2005.
[36] Huang, H., H. Feng, and C. Peng, Complete local Fisher discriminant analysis with Laplacian score ranking for face recognition. Neurocomputing, 2012. 89(10): p. 64-77.
[37] Cai, D., C. Zhang, and X. He. Unsupervised feature selection for multi-cluster data. in ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2010.
[38] Yang, S., et al., Unsupervised maximum margin feature selection via L 2,1 -norm minimization. Neural Computing & Applications, 2012. 21(7): p. 1791-1799.
[39] Ye, Q., C. Zhao, and N. Ye, Least squares twin support vector machine classification via maximum one-class within class variance. Optimization Methods & Software, 2012. 27(1): p. 53-69.