Feature Selection via Global Redundancy Minimization

De Wang, Feiping Nie, and Heng Huang

Abstract—Feature selection has been an important research topic in data mining, because the real data sets often have high-dimensional features, such as the bioinformatics and text mining applications. Many existing filter feature selection methods rank features by optimizing certain feature ranking criterions, such that correlated features often have similar rankings. These correlated features are redundant and don't provide large mutual information to help data mining. Thus, when we select a limited number of features, we hope to select the top non-redundant features such that the useful mutual information can be maximized. In previous research, Ding et al. recognized this important issue and proposed the minimum Redundancy Maximum Relevance Feature Selection (mRMR) model to minimize the redundancy between sequentially selected features. However, this method used the greedy search, thus the global feature redundancy wasn't considered and the results are not optimal. In this paper, we propose a new feature selection framework to globally minimize the feature redundancy with maximizing the given feature ranking scores, which can come from any supervised or unsupervised methods. Our new model has no parameter so that it is especially suitable for practical data mining application. Experimental results on benchmark data sets show that the proposed method consistently improves the feature selection results compared to the original methods. Meanwhile, we introduce a new unsupervised global and local discriminative feature selection method which can be unified with the global feature redundancy minimization framework and shows superior performance.

Index Terms—Feature selection, feature ranking, redundancy minimization

1 Introduction

Recent fast improvements in research and ongoing developments in information technology enables us to collect huge amounts of data. Analyzing these big data has become a key basis of competition, underpinning new waves of productivity growth, innovation, and consumer surplus. Many data mining and machine learning approaches have been developed to analyze and understand the scientific data for different applications. Among them, feature selection is one of most important techniques, and can enhance other data mining tasks, such as classification and clustering.

Feature selection is to select relevant and informative features from the high-dimensional space, and plays a crucial role in many scientific and practical applications, because it can speed up the learning process, improve the mode generalization capability, and decrease the algorithm running time in the real applications. The wide range of application areas include computational neuroscience [34], imaging genomics [36], [37], protein property prediction [35], image annotation [7], text mining [21], etc.

The high-dimensional data mining often suffer the *curse of dimensionality* issue [23], which is often caused by the high-dimensional features and a small size of samples. For example, in The Cancer Genome Atlas (TCGA) data mining, each sample has several ten thousand gene expression measures and millions of single nucleotide polymorphism (SNP) measures, however the cancers or the subtypes of cancers are only associated with a small set of genes or SNPs. In protein sequence analysis, the nucleotide or amino acid at each position in a sequence or the higher order combinations of these building blocks (k-mer) are often used as features, but many of them are irrelevant to the sequence analysis tasks and confuse the analysis process. In proteomics, the Mass spectrometry is often used for disease diagnosis and protein-based biomarker profiling, but each mass spectrum sample can contain up to 15,500 intensity values as features in low-resolution MALDI-TOF data (matrix-assisted laser desorption/ionization-time of flight, which has a large mass range). Although we can employ the traditional dimension reduction method, i.e. principle component analysis (PCA), linear discriminant analysis (LDA) and etc., to reduce the feature size, we cannot tackle the problems where the features have natural meanings and cannot be numerically combined, such as text mining [11], to select text key words; DNA microarray [27], to find out a few of genes associated with a given disease; mass spectrometry [31], to discover the protein-based biomarker profiling [31]. Therefore, feature selection is an essential component of data mining research and a large number of developments on feature selection have been made in the literature [13], [14].

There are three types of feature selection methods: filter method [8], [17], [20], [27], [29], wrapper method [19], and embedded method [5], [6], [26], [32]. The filter methods have low computational cost, but the selected features often cannot achieve good classification performance. The features selected by wrapper methods usually

• *The authors are with the Department of Computer Science and Engineering, University of Texas at Arlington, Arlington, TX 76019. E-mail: {wangdelp, feipingnie}@gmail.com, heng@uta.edu.*

Manuscript received 10 July 2014; revised 26 Jan. 2015; accepted 8 Apr. 2015. Date of publication 27 Apr. 2015; date of current version 8 Sept. 2015. Recommended for acceptance by S. Yan. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below. Digital Object Identifier no. 10.1109/TKDE.2015.2426703

have good performance. However, the wrapper methods use classification results to select feature, hence their computational cost is very high and is not suitable for large-scale applications. The embedded methods incorporate feature search and classification model into a single optimization problem, and usually is faster than the wrapper methods and slower than the filter methods.

Although there are many different types of feature selection approaches, their mechanism is the same, i.e. all of them use different ways to rank features, such as score function, classification results, weights from the model parameter matrix. Correlated features usually tend to get similar rankings, because they are considered equally important for classification. As a result, many top ranked features are often correlated to each other. From statistics point of view, these correlated features are redundant and more redundant features may not introduce extra useful information to help data mining. Thus, when we select a limit number of features, we hope to select the top non-redundant features such that the useful mutual information can be maximized.

Feature redundancy can be measured by the cosine similarity between features. We use global redundancy to represent the sum of redundancy of a feature with all other features. In previous research, Ding et al. recognized this important issue and proposed the minimum Redundancy Maximum Relevance Feature Selection (mRMR) model [8], [27] to minimize the redundancy between sequentially selected features. However, such a redundancy minimization way is a greedy search, thus the global redundancy wasn't considered and the feature selection results are not optimal.

To address this problem, we propose a new feature selection framework to globally minimize feature redundancy with maximizing the ranking scores. The feature ranking results of any feature selection method can be used as the input of our new framework. The proposed model minimizes the global redundancy and refines the ranking scores, such that the feature rankings are improved. Our new framework can be applied to both unsupervised and supervised feature selection methods. Experimental results on benchmark data sets show that the proposed global redundancy minimization (GRM) framework consistently improves the feature selection results compared to the original methods.

In the real world data mining applications, labeling data is time and labor consuming, thus we often have a large number of unlabeled data and a small number of labeled data. Although the supervised feature selection methods suppose to get better results than the unsupervised ones due to the utilizations of label information, when the number of labeled data is small and cannot represent the true data distribution, the supervised methods may select the incorrect features and lead to the worse results than the unsupervised ones. In our experiments, we will also demonstrate this key point. Thus, the unsupervised feature selection methods are crucial for practical applications. There are many unsupervised feature selection algorithms developed. For example, TRACK [33] proposed an unsupervised feature selection algorithm via iteratively perform trace ratio minimization and K-means clustering.[4] proposed an unsupervised feature selection algorithm via regression on eigen-vector of Laplacian graph. In this paper, we will also introduce a new unsupervised feature selection method, which simultaneously considers global variance and local variance, such that the selected features are both global and local discriminative. Combined with our proposed global feature redundancy minimization framework, the new feature selection method outperforms the related methods in most experimental results.

2 A New Framework for Improving Feature Ranking via Global Redundancy Minimization

As the goal of feature selection is to select a compact subset of features to represent data, we expect the selected features can provide maximal mutual information with response/target variable. If feature i and feature j are highly correlated, i.e. the absolute value of correlation coefficient $|\rho_{ij}|$ is large, it is preferable to retain one feature and ignore the other one for compactness. Because the retained feature can represent most variance resulted from the two features. When the number of selected features is fixed, the selected features with less redundancy can provide more mutual information and show larger discriminative power. Thus, redundant features should be eliminated during the feature selection process, such that the optimal and compact subset of features can be selected.

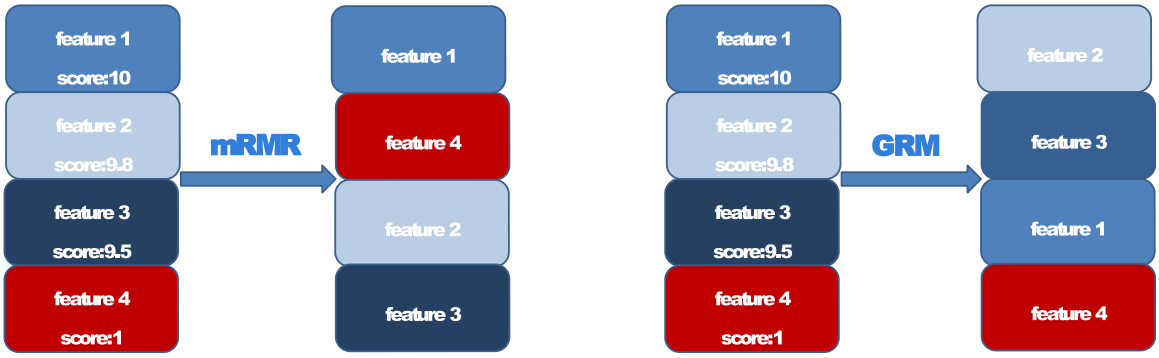
Ding et al. [8], [27] proposed a greedy approach (mRMR) to eliminate the redundant features: if one feature f_i is selected, other features highly correlated with f_i are excluded. However, these excluded features may have low global redundancy in all, and lead to the non-optimal feature selection results.

For example, there are four features with ranking scores: 10, 9.8, 9.5, 1 (higher score means more discriminative), as shown in Fig. 1. The ranking score of a feature is essentially determined by the correlation between the feature and the response variable. For the aforementioned example, the second and the third features are uncorrelated: the second feature explains one aspect of variation in response variable and the third feature explain another aspect of variation. The first feature is correlated with the second and the third features, and explains most of the variations in response variable, therefore highest score. However, the combination of feature 2 and feature 3 contains more useful information than feature 1, therefore, it is better to select feature 2 and feature 3.

If we select two features from them, the mRMR method will select the features with scores 10 and 1. However, the ideal result is to select two features with scores 9.8 and 9.5. But these two features have high redundancy with the first feature, and are missed by the greedy algorithm. Thus, instead of greedily minimizing the redundancy, we should minimize the global feature redundancy. To tackle this problem, in this paper, we propose a new feature selection model for improving feature ranking via global redundancy minimization.

2.1 GRM Framework Formulation

Most existing feature selection methods select features by ranking features, i.e. utilizing different criterions (such as filter function, classification accuracy, weight in coefficient matrix) to compute a score for each feature such



(a) mRMR (b) GRM

Fig. 1. Illustration of mRMR (left) and GRM (right) mechanism. The first feature is correlated with the second and the third features, but the second and the third features are uncorrelated.

that all features can be ranked based on scores. None of these methods is universally better than others, especially in the practical applications. Thus, our new global redundancy minimization framework is designed to integrate any types of feature ranking scores. Given the set of feature ranking scores, the GRM model will reduce the global feature redundancy and keep the ranking consistence by solving the following objective:

$$\min_{z^T\mathbf{1}=1, z\geq 0} \frac{z^T A z}{z^T s}, \tag{1}$$

where $s \in \Re^d$ (d is the number of features) is the input feature score vector computed from a type of feature ranking criterion (e.g. our new feature selection method introduced in next section); $z \in \Re^d$ is the new feature score vector resulted by our GRM model; $A \in \Re^{d\times d}$ is the redundancy matrix including the cosine similarity based correlations between features, which can be obtained by:

$$A_{ij} = B_{ij}^2 = \left(\frac{{f_i}^T f_j}{\|f_i\|\|f_j\|}\right)^2, \tag{2}$$

where $f_i \in \mathbb{R}^{n\times 1}$, $f_j \in \mathbb{R}^{n\times 1}$ are column vectors of the ith and jth features after centerization (*i.e.* mean value is zero). Because both high positive and negative correlations represent the high redundancy between features, so we need to also penalize on high negative correlation. One may come up with using the absolute value of negative correlation, *i.e.* using $|B|$. However, it is not guaranteed that $|B|$ is positive semi-definite when B is positive semidefinite. So in this paper, we use the squared cosine similarity to measure the feature redundancy. We will show later that by using squared cosine similarity, the matrix is guaranteed to be positive semi-definite. With this property, we can iteratively solve a convex formulation to optimize the proposed objective in Eq. (1).

The numerator $z^T A z$ in the objective function (1) represents global feature redundancy in the refined feature ranking. The denominator $z^T s$ is the consistency between the refined feature score z and original score s. By minimizing Eq. (1), the global redundancy is minimized, and the consistency between the refined feature score z and original score s is maximized. If the feature f_i has high global redundancy, the first term will dominant the minimization and its score z_i will get small value such that the ranking is low. On the other hand, if the feature f_i has low global redundancy, the second term will dominant the minimization and its score z_i will be similar to the original score s_i such that the ranking won't be reduced. Note that our model is parameter free, so the time needed for applying the GRM framework is very little (it takes less than 1 minute in the experiment).

In literature, we found the other independent work [30] proposed to solve a similar problem in order to achieve minimum redundancy. However, in [30], there is one parameter need to be tuned. The model proposed in our work is parameter free. More important, the correlation matrix used in [30] may not be positive semi-definite, which lead to a non-convex problem, and the global optimum can not be obtained. In contrast, the positive semi-definiteness of the correlation matrix is guaranteed in our work.

With applying our model, features which are redundant from a global point of view (captured by redundancy matrix A) are expected to get lower refined score and thus be excluded from the selected feature subset. The constraint $z^T\mathbf{1} = 1$ is important and avoids the trivial minimization on all features scores, which is caused by the values 1 on the diagonal of matrix A (*i.e.* features have the largest redundancy with themselves).

2.2 Positive Semi-Definiteness of the Similarity Matrix

The cosine similarity matrix B is positive semidefinite. Our following Lemma will show that the matrix $B \circ B$ ($\circ$ is the element product) is also positive semidefinite.

Lemma 2.1. *If the matrices X and Y are positive semidefinite with the same size, the matrix $X \circ Y$ is also positive semidefinite.*

Proof: Because X and Y are positive semidefinite, there are U and V satisfy that $X = UU^T = \sum_i u_i u_i^T$ and $Y = VV^T = \sum_i v_i v_i^T$. Thus,

$$X \circ Y = \sum_{i,j} (u_i u_i^T) \circ (v_j v_j^T). \tag{3}$$

We are going to prove the matrix $(uu^T) \circ (vv^T)$ is positive semidefinite for any u and v. For any vector x, we have

$$\begin{aligned} & x^T((uu^T) \circ (vv^T))x \\ &= \sum_{i,j} x(i)x(j)u(i)u(j)v(i)v(j) \\ &= \sum_i x(i)u(i)v(i) \sum_j x(j)u(j)v(j) \geq 0\,. \end{aligned} \tag{4}$$

Thus, $(uu^T) \circ (vv^T)$ is positive semidefinite. Based on this result, we can conclude $X \circ Y = \sum_{i,j}(u_i u_i^T) \circ (v_j v_j^T)$ is positive semidefinite. □

Based on Lemma 1, we know the redundancy matrix $A = B \circ B$ is positive semi-definite, thus our new objective in (1) is convex. Thus, the converged optimization solution is the global solution.

Note that our new framework can be applied to the ranking scores of both supervised and unsupervised feature selection methods, as long as the methods define a ranking score for each feature.

2.3 Optimization Algorithm Solving GRM Model

We use augmented lagrangian multiplier (ALM) method [3], [16] to solve problem (1). The convergence of ALM algorithm was proved and discussed in previous papers [3], [16], [22], [28]. Here we skip this discussion.

2.3.1 Standard Augmented Lagrangian Multiplier

Consider the following general constrained optimization problem:

$$\min_{h(X)=0} f(X). \tag{5}$$

We can use ALM to solve the problem as described in Algorithm 1.

Algorithm 1. Algorithm for probleZm (5)

Set $1 < \rho < 2$. Initialize $\mu > 0, \Lambda$
repeat
1. Update X by $\min_X f(X) + \frac{\mu}{2}\left\|h(X) + \frac{1}{\mu}\Lambda\right\|_F^2$
2. UpdateΛ by $\Lambda = \Lambda + \mu h(X)$
3. Update μ by $\mu = \rho\mu$

until Converges

2.3.2 Solving Problem (1) Using ALM

Problem (1) can be solved by iteratively solving the following problem:

$$\min_{z^T\mathbf{1}=1, z\geq 0, z=v} v^T A v - \lambda v^T s, \tag{6}$$

where $\lambda = \frac{z^T A z}{z^T s}$ is calculated using z from last iteration.

According to Step 1 in Algorithm 1, it is equivalent to solve the following problem:

$$\min_{z^T\mathbf{1}=1, z\geq 0} v^T A v - \lambda v^T s + \frac{\mu}{2}\left\|z - v + \frac{1}{\mu}\alpha\right\|_F^2. \tag{7}$$

Minimizing Eq. (7) jointly with respect to v, s is difficult, so we alternately update one variable each time while fix another variable.

When we fix z, problem (7) is simplified into solving an unconstrained optimization problem with respect to v. Set the derivative of Eq. (7) with respect to v as zero, we get:

$$v = \left(A + \frac{\mu}{2}I\right)^{-1}\left(\frac{\lambda}{2}s + \frac{\mu}{2}(z + \frac{1}{\mu}\alpha)\right). \tag{8}$$

When we fix v, problem (7) is simplified into solving the following problem:

$$\min_{z^T\mathbf{1}=1, z\geq 0}\left\|z - v + \frac{1}{\mu}\alpha\right\|_F^2. \tag{9}$$

In [12], an efficient piecewise root finding (PRF) algorithm has been proposed to solve euclidean projection problems on with ℓ_1 ball constraint, Elastic Net (EN) constraint and Intersection of a Hyperplane and a Halfspace (IHH) constraint. Problem (9) is essentially a euclidean projection on the simplex space. In the next section, we derive the optimization algorithm for problem (9).

2.3.3 Solution for Problem (9)

Take $w = v - \frac{1}{\mu}\alpha$, problem (9) is equivalent to:

$$\min_{z^T\mathbf{1}=1, z\geq 0} \frac{1}{2}\|z - w\|_F^2. \tag{10}$$

The Lagrangian function of problem (10) is:

$$\frac{1}{2}\|z - w\|_F^2 - \gamma z^T\mathbf{1} - \lambda^T z, \tag{11}$$

where γ and λ are Lagrangian coefficients. Denote the optimal solution of problem (10) as z^*, and the corresponding Lagrangian coefficients are γ^* and λ^*. We can get the following equations according to KKT condition:

$$\begin{cases} \forall i, \quad z_i^* - w_i - \gamma^* - \lambda_i^* = 0, & (12) \\ \forall i, \quad z_i^* \geq 0, & (13) \\ \forall i, \quad \lambda_i^* \geq 0, & (14) \\ \forall i, \quad z_i^*\lambda_i^* = 0. & (15) \end{cases}$$

We can rewrite Eq. (12) as:

$$z^* - w - \gamma^*\mathbf{1} - \lambda^* = 0. \tag{16}$$

Multiply both side with $\mathbf{1}^T$ and use the constraint $z^T\mathbf{1} = 1$, we get:

$$\gamma^* = \frac{1 - \mathbf{1}^T v - \mathbf{1}^T\lambda^*}{n}. \tag{17}$$

Substitute Eq. (17) into (16), we get:

$$z^* = \left(w - \frac{\mathbf{1}\mathbf{1}^T}{n}w + \frac{1}{n}\mathbf{1} - \frac{\mathbf{1}^T\lambda^*}{n}\mathbf{1}\right) + \lambda^*. \tag{18}$$

Denote $\bar{\lambda}^* = \frac{\mathbf{1}^T\lambda^*}{n}$, and $u = w - \frac{\mathbf{1}\mathbf{1}^T}{n}w + \frac{1}{n}\mathbf{1}$, Eq. (18) can be written as:

$$z^* = u + \lambda^* - \bar{\lambda}^*\mathbf{1}. \tag{19}$$

So $\forall i$, we have:

$$z_i^* = u_i + \lambda_i^* - \bar{\lambda}^*. \tag{20}$$

According to Eqs. (13)-(15) and Eq. (20), we know:

$$u_i + \lambda_i^* - \bar{\lambda}^* = (u_i - \bar{\lambda}^*)_+. \tag{21}$$

So we get:

$$z_i = (u_i - \bar{\lambda}^*)_+. \tag{22}$$

Therefore,we can get the optimal solution if we know the value of $\bar{\lambda}^*$.

Eq. (20) can be written as: $\lambda_i^* = z_i^* - \bar{\lambda}^* - u_i$. Also, we know $\lambda_i^* = (\bar{\lambda}^* - u_i)_+$ according to Eqs. (13)-(15). Hence we get:

$$\bar{\lambda}^* = \frac{1}{n}\sum_{i=1}^{n}(\bar{\lambda}^* - u_i)_+. \tag{23}$$

We define the following function:

$$f(\bar{\lambda}) = \frac{1}{n}\sum_{i=1}^{n}(\bar{\lambda} - u_i)_+ - \bar{\lambda}. \tag{24}$$

So $\bar{\lambda}^*$ is the root the $f(\bar{\lambda}) = 0$. The root can be found using Newton method easily. Then we can get the optimal solution according to Eq. (22).

After the above analysis, we conclude the detailed algorithm for solving problem (1) in Algorithm 2.

Algorithm 2. Algorithm for Problem (1)

Initialize z
repeat
  Set $\lambda = \frac{z^T A z}{z^T s}$
  Set $1 < \rho < 2$.
  Initialize $\mu > 0, \alpha$
  repeat
    1. Update v by Eq. (8)
    2. Update z by Eq. (22)
    3. Update α by $\alpha = \alpha + \mu(z - v)$
    4. Update μ by $\mu = \rho\mu$
  until Converges
until λ converges to minimum
Output z

Algorithm 3. Procedure of GRM Framework

Input: data matrix X data label Ynumber of selected features k

1. Construct the similarity matrix A using a similarity measure (squared cosine similarity, mutual information, sparse representation of a feature)
2. Compute the feature score s using a supervised or unsupervised feature selection algorithm
3. Applying the GRM framework by solving the objective function (1) using Algorithm 2, get the refined feature score z after eliminating feature redundancy
4. Ranking features according z, select the top k features

Output: top k features

The main computational cost is dominated by computing the inverse of the $d \times d$ correlation matrix in Eq. (8), which cost $O(d^3)$. So the total computational cost is $O(Td^3)$. This seems a little bit overwhelming if d is large. However, note that we do not need to use all features since our goal is to get the top k features. We can use only the top subset of features ranked by the original feature score s, so we can get the target top k features very fast.

The procedure for applying the GRM framework is summarized in Algorithm 3.

2.4 Feature Selection with Prior Knowledge Constraints

In practical applications, we often have domain or prior knowledge on features. For example, in many bioinformatics problems, some features have synthetic effect in controlling a certain disease. Those features should belong to the same group, and should be chosen together. Once we know which features belong to the same group, we can incorporate the group information in our model. We use a new matrix $G \in \Re^{g \times d}$ (d is the number of features, $g = d * (d-1)/2$ is the number of feature pair) to keep the feature group information. If features i and j (i and j are feature ranking indices based on the feature score vector s) are in the same group with the feature pair index k, $G_{ki} = 1$ and $G_{kj} = -1$. Our new objective is to solve:

$$\min_{z^T\mathbf{1}=1, z\geq 0} \frac{z^T A z + \gamma ||Gz||_2^2}{z^T s}, \tag{25}$$

where if two features are in the same group, the minimization of $||Gz||_2^2$ will make their scores similar. The above objective can be written as:

$$\min_{z^T\mathbf{1}=1, z\geq 0} \frac{z^T (A + \gamma G^T G) z}{z^T s}. \tag{26}$$

It is of the same form of problem (1), so we can use a similar algorithm outlined in Algorithm 4 to solve this new objective.

Algorithm 4. Algorithm for Problem (25)

Initialize z
repeat
  Set $\lambda = \frac{z^T (A + \gamma G^T G) z}{z^T s}$
  Set $1 < \rho < 2$.
  Initialize $\mu > 0, \alpha$
  repeat
    1. Update v by:
    $v = (A + \gamma G^T G + \frac{\mu}{2} I)^{-1} (\frac{\lambda}{2} s + \frac{\mu}{2}(z + \frac{1}{\mu}\alpha))$
    2. Update z by Eq. (22)
    3. Update α by $\alpha = \alpha + \mu(z - v)$
    4. Update μ by $\mu = \rho\mu$
  until Converges
until λ converges to minimum
Output z

3 A New Unsupervised Feature Selection Algorithm

Besides proposing the novel feature ranking improvement framework, we are also going to introduce a new efficient unsupervised feature selection approach, which can be combined with the above framework as a unified feature selection procedure for practical applications. First, we briefly revisit a commonly used unsupervised feature selection algorithm: Laplacian Score. After that, we propose a

new unsupervised feature selection criterion with both local and global discriminations.

3.1 Laplacian Score

One commonly used unsupervised feature selection criterion is Laplacian Score [15], which aims to select features that can best preserve the local manifold structure. He et al. [15] argued that: in many classification tasks, the local structure of data is expected to be more important than global structure.

Motivated by such observation, Laplacian Score was proposed to capture the local structure of data using graph Laplacian. The Laplacian Score of a feature is defined as:

$$s(k) = \frac{f_k^T L f_k}{f_k^T D f_k}, \tag{27}$$

where f_k is the kth feature, $L = D - W$ is the graph Laplacian matrix, W is often computed using a heat kernel as following:

$$w_{i,j} = exp(-\frac{\left\|x_i - x_j\right\|^2}{2\sigma^2}), \tag{28}$$

where $w_{i,j}$ is the similarity between the ith sample and the jth sample, σ is the heat kernel parameter.

Laplacian score method need to construct a similarity matrix using heat kernel beforehand. One drawback of the Laplacian Score strategy is that: it is very sensitive to the heat kernel parameter σ which is used to construct similarity matrix. Thus, a huge amount of computation is needed to tune the parameter σ. On the other hand, constructing the similarity matrix itself is also time consuming (the time complexity is at least $\mathcal{O}(dn^2)$).

3.2 Unsupervised Local and Global Discriminative (LGD) Feature Selection

The Laplacian score only considers the local structure. In order to utilize both local and global structures, we propose an unsupervised local and global discriminative feature selection criterion. The score of each feature is defined as the ratio between global variance and local variance:

$$s(k) = \frac{\sum_i (f_k(i) - \bar{f}_k)^2}{\sum_j \sum_{f_k(i) \in o(f_k(j))} (f_k(i) - \bar{f}_{kj})^2}, \tag{29}$$

where $f_k(i)$ is the kth feature of the ith sample, $\bar{f}_k$ is the mean of the kth feature, $o(f_k(j))$ is the set of neighbor points of the jth sample. $\bar{f}_{kj}$ is the mean computed from the set of neighbour points as:

$$\bar{f}_{kj} = \frac{\sum_{f_k(i) \in o(f_k(j))} f_k(i)}{|o(f_k(j))|}, \tag{30}$$

where $|o(f_k(j))|$ is the cardinality of the neighbor set. Because the neighbor points of point $f_k(i)$ can be efficiently computed by sorting the values $f_k(1), \ldots, f_k(n)$, the computational complexity of our method is $O(dn\ log\ n)$.

The proposed LGD criterion prefers to select features with large global variance and small local variance. Such features are expected to be discriminative for classification/clustering tasks.

It is deserved to mention that the only parameter in our proposed criterion is the size of the neighbourhood set, i.e. the number of data points in the set. If the parameter is too small, local variance is not captured well enough. If the parameter is too large, the boundary between local and global blurs. This parameter plays a similar role with the σ in Laplacian score. However, it is easier to be set in practice compared with σ. Empirically, the performance of the algorithm is pretty good by setting the size of neighbourhood sets somewhere between 5 to 8, as we will show in the experimental section. However, for the parameter σ, we do not have any prior information without looking at a specific dataset.

Unlike Laplacian Score in which a similarity graph must first be constructed in order to harness the discriminative power of the local structure in data, our method directly computes the local variance from the original data, which is more computationally efficient.

4 Experimental Results

4.1 Data Set Descriptions

We evaluate the performance of our GRM framework on extensive benchmark data sets as described below.

GLIOMA data set contains 50 samples, each of which comprises of 4,434 genes after excluding genes with minimal variation. Each sample belongs to one of the four subtypes of cancers: cancer glioblastomas (14 samples), non-cancer glioblastomas (14 samples), cancer oligodendrogliomas (seven samples) and non-cancer oligodendrogliomas (15 samples).

LUNG data set comprises of 203 samples. These samples are from five classes, which have 17, 6, 20, 21, 139 samples, respectively. 3,312 genes are retained after excluding genes whose standard deviation are smaller than 50 expression units.

AMLALL data set contains 72 samples, in which 47 subjects are acute lymphoblastic leukemia (ALL) and 25 subjects are acute myeloid leukemia (AML). Every sample contain 7,219 gene expression values.

ProCancer data set contains 89 samples, and each sample contain 15,154 genes expression values. Samples are from two classes: normal and prostate cancer.

LEU data set contains 72 samples of which some are leukemia patients. The data set has 3,571 gene descriptors.

small round blue cell tumors (SRBCT) contains 83 samples with 2,308 genes: 29 cases of Ewing sarcoma (EWS), coded 1, 11 cases of Burkitt lymphoma (BL), coded 2, 18 cases of neuroblastoma (NB), coded 3, 25 cases of rhabdomyosarcoma (RMS), coded 4. A total of 63 training samples and 25 test samples are provided in [18].

Columbia object image library (COIL20) [25] is a data set of gray-scale image of 20 objects. The objects were put on a truntable, and each object contains 72 images which is rotated 5 degree apart each time.

BINALPHA data set [2] contains images of handwritten alphabets.

DIGIT data set is a public data set hosted in UCI Machine Learning Repository.[1] This data set consists of handwritten digits from 0 to 9, and each digit is a class. There are

1. https://archive.ics.uci.edu/ml/datasets/Multiple+Features

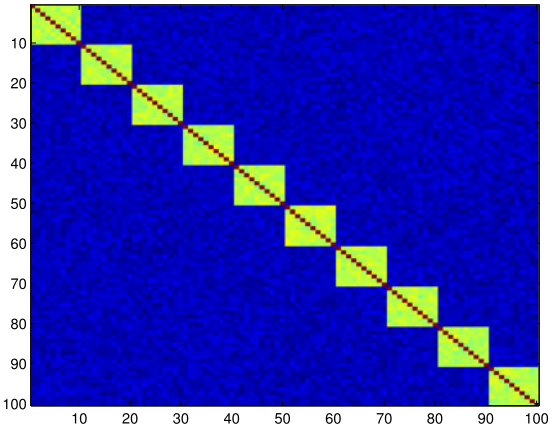


Fig. 2. Feature covariance matrix. Features within the same group are highly correlated.

200 patterns for each digit, thus 2,000 samples in total. Each sample is represented by six types of features as following:

1. FOU features, which consist of 76 Fourier coefficients of the character;

2. FAC features, which consist of 216 profile correlations;

3. KAR features, which consist 64 Karhunen-love coefficients;

4. PIX features, which consist of 240 average pixel values of 2×3 windows;

5. ZER features, which consist of 47 Zernike moments; 6. MOR features, which consist of six morphological features.

CALTECH data set contains object images coming from different categories. We extracted different types of features (LBP [1], HOG [38], GIST [9], SIFT [24], CENTRIST [39]) from the original image.

We also do experiments on synthetic data set which is generated in the following way: We generate 400 samples with 100 features. The 100 features are from 10 groups, each group contains 10 features which are highly correlated. Fig. 2 shows the covariance matrix of features, each group forms a block in the diagonal. We then use this covariance matrix to generate two clusters of points drawn from Gaussian distribution.

We summarize the data sets in Table 1.

TABLE 1
Data Sets Description

	Class #	Sample #	Feature #
Synthetic	2	400	100
GLIOMA	4	50	4,434
LUNG	5	203	3,312
AMLALL	2	72	7,219
ProCancer	2	89	15,154
LEU	2	72	3,571
SRBCT	4	83	2,308
COIL20	20	1,440	1,024
BINALPHA	36	1,404	3,20
PalmData	100	2,000	256
DIGIT	10	2,000	649
CALTECH	7	441	2,218

4.2 Experiment Setup

In order to validate the effectiveness of our GRM framework, we apply it to some unsupervised and supervised feature selection criterion. Unsupervised feature selection algorithm used are Laplacian Score (Lap), MCFS [4], and our proposed LGD criterion. Feature selection after applying GRM framework are denoted as GLap, GMCFS, GLGD, respectively.

Laplacian score aims to select features that can best preserve the local manifold structure. Our proposed LGD method considers both global variance and local variance simultaneously. So we compare our method with Laplacian score to show the superiority of our method.

MCFS is another popular unsupervised feature selection method. It first performs spectral decomposition on the Laplacian matrix, then uses the decomposed eigenvectors as response vector for regression. Features with larger regression coefficients are selected. MCFS has been reported to achieve very promising results, so we compare our method with this method.

To demonstrate the effectiveness of unsupervised feature selection technique in the case of label scarsity, we also compare them with RF algorithm using only part (20 percent) of the labeled data for feature selection, denoted as RF (20 percent).

For supervised feature ranking criterion, we apply the GRM framework on ReliefF (RF) [20], Fisher score (Fscore) [10] and information gain (IG) [29]. Fisher score is one of the most simple but effective supervised feature selection method. Features with large between-class distance and small within-class distance are preferred by Fisher score method. ReliefF is another representative filter method. The weight of each feature increases if it differs from nearby instances of the other class more than nearby instances of the same class. Information gain measures the reduction of entropy for each feature.

For each feature selection method, five-fold cross-validation is used to evaluate the performance using two popular classifiers: linear SVM and KNN.

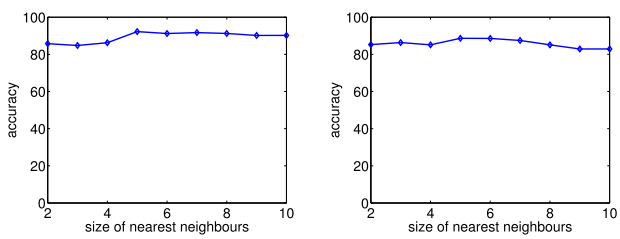
We define the following evaluation metric to measure the redundancy in selected features:

$$Red(\mathbf{S}) = \frac{1}{m(m-1)} \sum_{f_i, f_j \in \mathbf{S}, i \neq j} A_{i,j}, \tag{31}$$

where $\mathbf{S}$ is the set of selected features, m is the number of selected features, $A_{i,j}$ is the squared cosine similarity defined in the GRM framework.

4.3 Parameter Discussion of LGD Method

The parameter of number of nearest neighbours in LGD method defines the scope of locality. Fig. 3 shows the LGD feature selection performance with different number of



(a) LUNG (b) ProCancer

Fig. 3. LGD feature selection performance with different number of nearest neighbours.

TABLE 2
Classification Accuracy Using Top 20 Features Selected by Different Algorithm

	Lap	**GLap**	LGD	**GLGD**	MCFS	**GMCFS**	RF (20%)
Synthetic	94.75	96.25	94.50	96.25	95.50	**96.75**	94.75
GLIOMA	50.00	54.00	54.00	64.00	56.00	**66.00**	58.00
LUNG	77.42	90.64	79.88	**93.21**	83.85	91.24	83.28
AMLALL	65.18	70.71	76.25	**88.93**	65.00	70.18	86.46
ProCancer	82.88	84.54	80.78	**86.34**	80.72	85.16	85.16
LEU	83.39	**97.32**	91.46	93.25	93.04	93.04	92.86
SRBCT	49.33	54.04	50.12	**78.67**	60.10	65.06	75.27
COIL20	57.23	74.62	67.63	**82.74**	80.03	81.31	53.44
BINALPHA	37.49	42.14	36.72	47.88	45.47	**50.45**	25.11
PalmData	94.65	97.50	95.00	**97.75**	96.45	97.50	62.06
DIGIT	94.50	95.90	93.65	**96.90**	96.55	96.70	83.78
CALTECH	55.33	75.28	56.24	77.31	73.02	**80.51**	74.04

Five-fold cross-validation is performed using linear SVM.

nearest neighbours(Since space is limited, we only show the performance on two data sets). From the figure we can see that: the best performance is generally achieved by setting the number of nearest number somewhere between 5 to 8. For simplicity, we set the number of nearest neighbour as 5 without fine tuning.

4.4 GRM Framework for Unsupervised Feature Selection

Different unsupervised feature selection methods are used to select top 20 features. The classification accuracy of selected features are reported in Tables 2 and 3, and the redundancy of selected features are reported in Table 4.

4.4.1 Redundancy Comparison

From Table 4 we can see that: on all data sets, the redundancy of selected features of GLap, GLGD and GMCFS are much lower than that of Lap, LGD and MCFS, respectively. We can also see that: features selected by MCFS are less correlated than Lap and LGD, thus the classification accuracy is also better.

4.4.2 Classification Accuracy Comparison

From Table 2 we can see that: on all data sets, the classification accuracy of features selected by GLap and GLGD are *significantly better* than that of Lap and LGD. The classification accuracy of features selected by MCFS is better than Lap and LGD. This is because features selected by these two methods are more redundant than MCFS as we can see in Table 4. For MCFS, the classification accuracy is also improved by GRM framework, but not as significant as Lap/LGD.

TABLE 4
Redundancy of Top 20 Features Selected by Different Algorithm

	Lap	**GLap**	LGD	**GLGD**	MCFS	**GMCFS**
Synthetic	0.224	0.103	0.200	0.117	0.203	0.144
GLIOMA	0.719	0.521	0.474	0.268	0.224	0.051
LUNG	0.651	0.198	0.617	0.093	0.106	0.023
AMLALL	0.657	0.336	0.273	0.062	0.147	0.020
ProCancer	0.976	0.385	0.695	0.383	0.281	0.024
LEU	0.404	0.148	0.184	0.045	0.148	0.024
SRBCT	0.247	0.071	0.144	0.036	0.132	0.029
COIL20	0.533	0.394	0.469	0.171	0.087	0.035
BINALPHA	0.129	0.029	0.128	0.020	0.028	0.012
PalmData	0.124	0.070	0.122	0.037	0.052	0.029
DIGIT	0.341	0.063	0.352	0.054	0.076	0.025
CALTECH	0.847	0.229	0.798	0.182	0.093	0.022

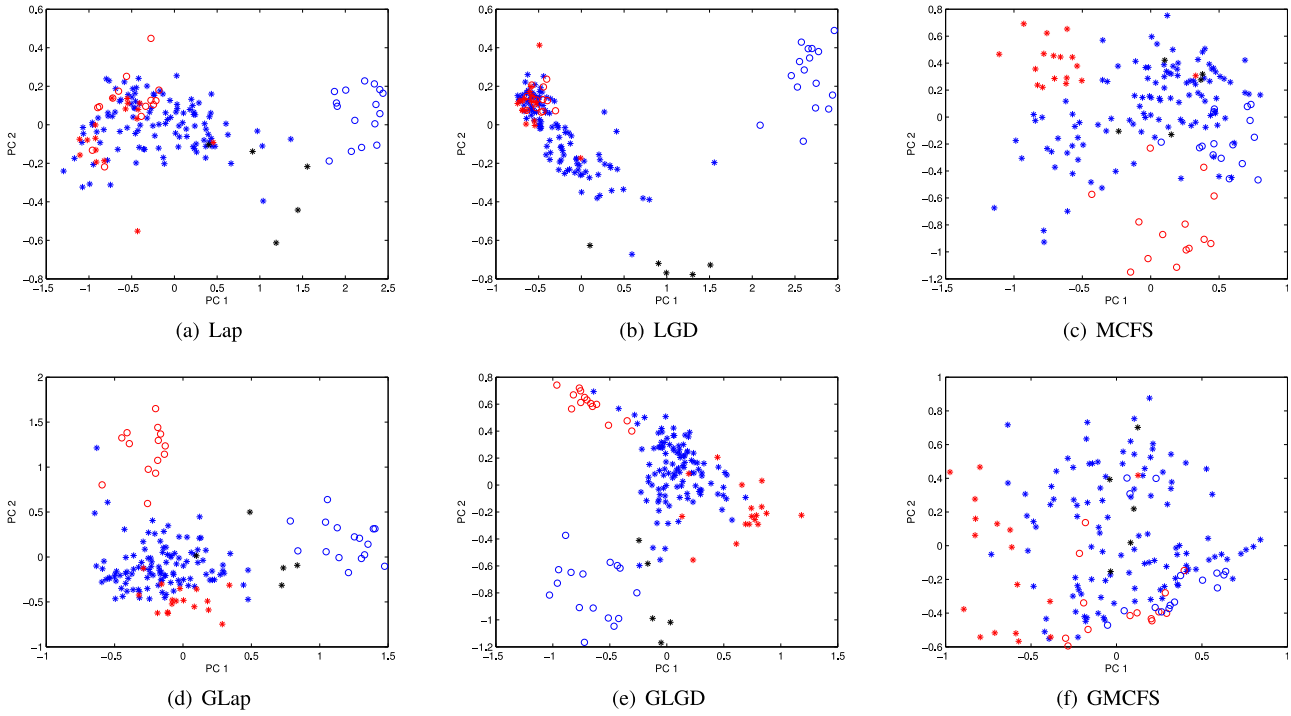
The improvement of classification accuracy shows the effectiveness of our GRM framework: By applying the framework, features which are redundant (from a global point of view) are expected to get lower refined score and thus be excluded from the selected subset of features. So the selected feature subset is compact and expected to be more discriminant as a whole when the size of the feature subset is fixed.

Table 2 also shows that: On all data sets, the best classification accuracy is achieved by the methods after applying the GRM framework. As for GLGD, it can achieve the best classification accuracy on most data sets. The best classification accuracy averaged on six data sets is also achieved by GLGD. This shows the effectiveness of our proposed unsupervised feature selection criterion with local and global discrimination. Although it tends to select correlated features, i.e. not compact, but the selected features are expected to be discriminant. So after applying the GRM framework, the selected feature subset became compact, and each single selected feature is discriminant, so the classification accuracy can outperform other methods.

TABLE 3
Classification Accuracy Using Top 20 Features Selected by Different Algorithm

	Lap	GLap	LGD	GLGD	MCFS	GMCFS	RF (20 Percent)
Synthetic	92.00	94.75	92.25	94.75	95.50	**96.00**	88.81
GLIOMA	56.00	68.00	60.00	**70.00**	64.00	64.00	52.50
LUNG	78.85	88.67	83.78	**93.14**	90.64	92.74	82.41
AMLALL	65.71	68.04	80.96	**81.79**	68.04	69.46	65.95
ProCancer	80.65	81.54	75.23	83.01	83.67	**85.16**	72.16
LEU	76.43	**83.21**	82.89	**83.21**	81.14	82.71	82.39
SRBCT	51.84	53.18	61.73	**77.02**	41.80	47.69	56.43
COIL20	68.58	77.06	75.69	80.21	80.56	**83.26**	57.07
BINALPHA	29.18	38.22	29.53	37.81	39.43	**43.23**	20.74
PalmData	94.90	96.05	95.00	**97.80**	96.20	97.40	62.46
DIGIT	94.85	95.30	94.15	**97.30**	95.55	96.10	77.79
CALTECH	64.62	74.59	65.32	78.23	75.97	**80.69**	73.75

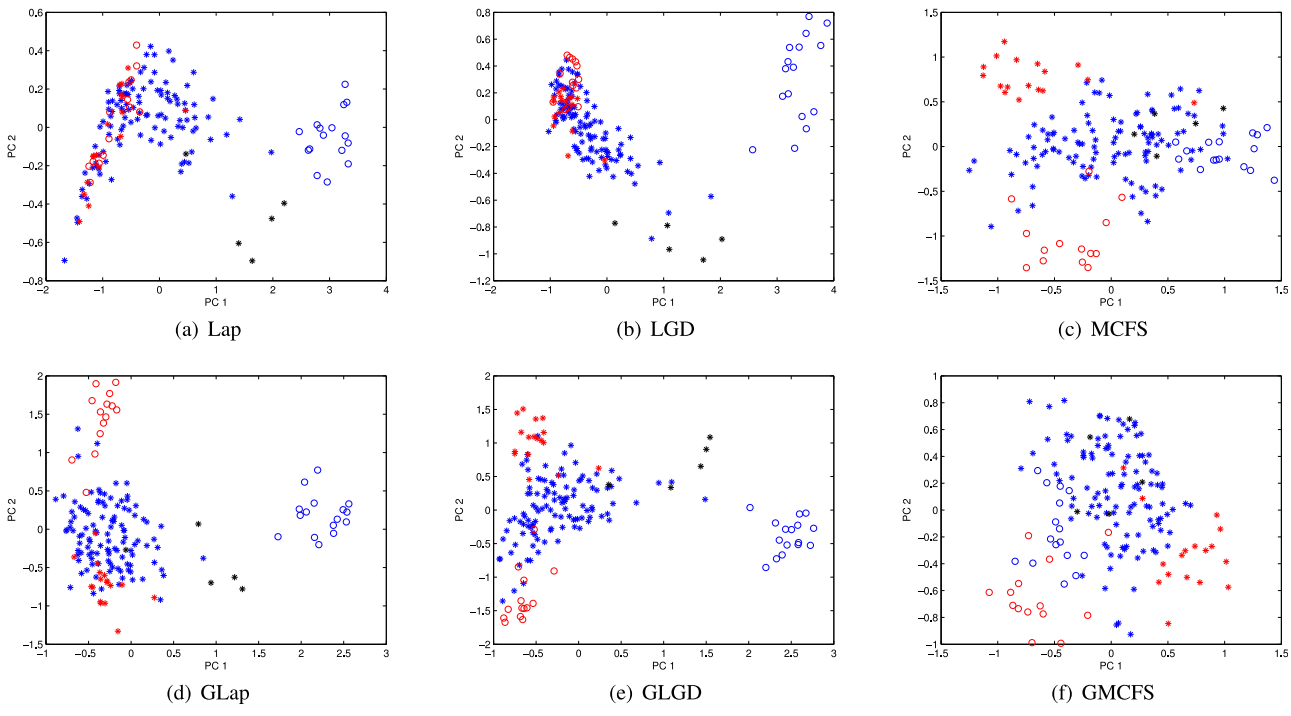
Five-fold cross-validation is performed using KNN.



(a) Lap (b) LGD (c) MCFS

(d) GLap (e) GLGD (f) GMCFS

Fig. 4. Projection on first two principle components using top 20 features for the LUNG data set. The horizontal axis is the score of the first principle component, and the vertical axis is the score of the second principle component. Different shape or color mark samples from different classes. Zoom in for clear view.



(a) Lap (b) LGD (c) MCFS

(d) GLap (e) GLGD (f) GMCFS

Fig. 5. Projection on first two principle components using top 40 features for the LUNG data set. The horizontal axis is the score of the first principle component, and the vertical axis is the score of the second principle component. Different shape or color mark samples from different classes. Zoom in for clear view.

The last column of Table 2 records the performance of ReliefF (20 percent). The performance of ReliefF (20 percent) is pretty good. This shows the importance of label information for feature selection: even though we only use 20 percent of labeled data for feature selection, the performance are comparable to unsupervised feature selection algorithms using all data on many data sets. From another point of view, however, we can achieve better classification accuracy by fully exploring huge amount of unlabeled data if labeled data are too scare, which justify the usefulness for developing unsupervised feature selection algorithms, especially in the coming big data era where data often comes in Terabyte but to label them may take a great amount of time and money.

TABLE 5
Classification Accuracy Using Top 20 Features Selected by Different Supervised Feature Selection Algorithm

	RF	GRF	Fscore	GFscore	IG	GIG	mRMR
Synthetic	94.75	95.00	94.25	**95.50**	94.25	95.00	85.75
GLIOMA	62.00	**68.00**	60.00	64.00	58.00	64.00	**68.00**
LUNG	78.42	89.67	78.42	**95.17**	78.42	95.64	78.42
AMLALL	85.89	88.93	91.61	**97.14**	93.04	95.54	93.04
ProCancer	91.05	93.33	93.14	**97.78**	94.21	96.67	88.63
LEU	94.46	95.21	92.57	93.04	95.71	**95.89**	94.14
SRBCT	85.73	88.08	95.29	96.47	96.47	**97.71**	96.74
COIL20	59.14	82.88	45.45	**85.47**	20.62	50.31	55.80
BINALPHA	25.26	44.05	29.20	**48.09**	25.71	44.47	27.13
PalmData	95.40	**98.00**	93.55	97.70	88.20	97.30	89.40
DIGIT	88.15	94.15	96.10	**97.15**	88.95	93.95	89.10
CALTECH	84.81	**88.67**	58.96	85.72	56.25	80.73	82.30

Five-fold cross-validation is performed using linear SVM.

TABLE 6
Classification Accuracy Using Top 20 Features Selected by Different Supervised Feature Selection Algorithm

	RF	GRF	Fscore	GFscore	IG	GIG	mRMR
Synthetic	92.50	94.00	92.00	**94.50**	93.00	93.75	84.50
GLIOMA	56.00	66.00	62.00	66.00	70.00	**74.00**	62.00
LUNG	86.81	91.67	84.17	93.35	93.17	**94.57**	90.81
AMLALL	80.54	90.18	83.04	91.61	84.82	**94.46**	**94.46**
ProCancer	86.67	88.76	96.67	**98.89**	94.44	94.44	91.96
LEU	84.64	94.46	95.89	95.89	91.61	**97.14**	**97.14**
SRBCT	78.90	91.61	93.41	95.29	79.53	**96.47**	95.82
COIL20	71.11	84.22	54.94	**86.03**	37.06	52.90	64.08
BINALPHA	19.29	41.41	21.37	**47.44**	22.23	39.10	22.90
PalmData	95.05	97.55	93.75	**97.85**	87.80	96.40	90.80
DIGIT	84.25	92.05	95.45	**96.65**	90.60	94.75	89.05
CALTECH	85.26	**87.08**	69.60	86.40	68.46	78.46	80.50

Five-fold cross-validation is performed using KNN.

4.4.3 Projection on First Two Principle Components (PC)

To visualize the discriminability of various feature selection algorithm, selected features are used to perform principle component analysis, then data samples are projected onto the first two principle components, as shown in Fig. 4 (PCA performed using top 20 features). We only show the projected figure for the LUNG data set since space is limited.

Figs. 4 and 5 show that: projected samples using features selected by GLap and GLGD are *much more separable* than that of Lap and LGD. The originally entangled three classes (red circle, red star and blue star) in Figs. 4c and 4e and Figs. 5c and 5e became much more separated from each other in Figs. 4d and 4f and Figs. 5d and 5f. This shows the effectiveness of our GRM framework: As the redundancy of selected features are pretty high for Lap and LGD, by applying GRM framework for redundancy reduction, the selected feature subsets became much more compact, and more discriminant as a whole when the size of feature subset is fixed.

The improvement of separability of projected samples using features selected by GMCFS is not as obvious as Lap/LGD. This is because features selected by MCFS is less redundant than Lap/LGD. However, the redundancy is further reduced by applying the GRM framework and the classification accuracy is improved as we can see from previous tables.

4.5 GRM Framework for Supervised Feature Selection

We apply the GRM framework on three supervised feature selection algorithms: ReliefF (RF), Fisher score (Fscore) and information gain. We also compare the GRM framework with mRMR [8], which is a well known feature selection algorithm for selecting non-redundant features. Redundancy of selected features are reported in Table 7, classification accuracy are shown in Tables 5 and 6.

4.5.1 Comparison of Redundancy and Classification Accuracy

From Table 7, we can see that: for all the three supervised feature selection algorithm, the redundancy of selected features became much lower after applying the GRM framework. The redundancy for features selected by Fscore is higher than other two methods. We can also notice that: the redundancy of mRMR is higher than GRF, GFscore, and GIG. This shows that the GRM framework can select a more compact subset of features than mRMR.

Tables 5 and 6 record the classification accuracy using various supervised feature selection algorithm. On all data sets, the classification accuracy has improved after applying the GRM framework. Especially on the GLIOMA and LUNG data sets, the improvement of classification accuracy is significant. Averaged on all data sets, the classification accuracy of mRMR is lower than GRF, GFscore, and GIG. The reason is that mRMR adopt an greedy approach to eliminate redundant features, so the selected feature subset is not global optimal. By contrast, the GRM framework take the global redundancy into consideration, thus the selected feature subset is expected to be more compact.

4.5.2 Projection on First Two Principle Components

Features selected by various supervised feature selection algorithms are used to perform PCA, then data samples are projected onto the first two PCs, as shown in Fig. 6 (PCA performed using top 20 features) and Fig. 7 (PCA performed using top 40 features). We only show the projected figure for the LUNG data set since space is limited. From

TABLE 7
Redundancy of Top 20 Features Selected by Different Supervised Feature Selection Algorithms

	RF	**GRF**	Fscore	**GFscore**	IG	**GIG**	mRMR
Synthetic	0.220	0.152	0.305	0.166	0.280	0.150	0.480
GLIOMA	0.459	0.088	0.567	0.273	0.463	0.200	0.288
LUNG	0.266	0.024	0.654	0.158	0.393	0.074	0.186
AMLALL	0.201	0.027	0.407	0.108	0.288	0.037	0.152
ProCancer	0.438	0.180	0.434	0.261	0.336	0.080	0.232
LEU	0.297	0.028	0.463	0.147	0.433	0.055	0.214
SRBCT	0.185	0.030	0.181	0.069	0.120	0.025	0.105
COIL20	0.576	0.161	0.390	0.149	0.248	0.044	0.110
BINALPHA	0.154	0.021	0.157	0.025	0.113	0.027	0.075
PalmData	0.114	0.034	0.133	0.036	0.156	0.034	0.110
DIGIT	0.196	0.031	0.266	0.056	0.205	0.037	0.301
CALTECH	0.307	0.037	0.879	0.146	0.880	0.160	0.175

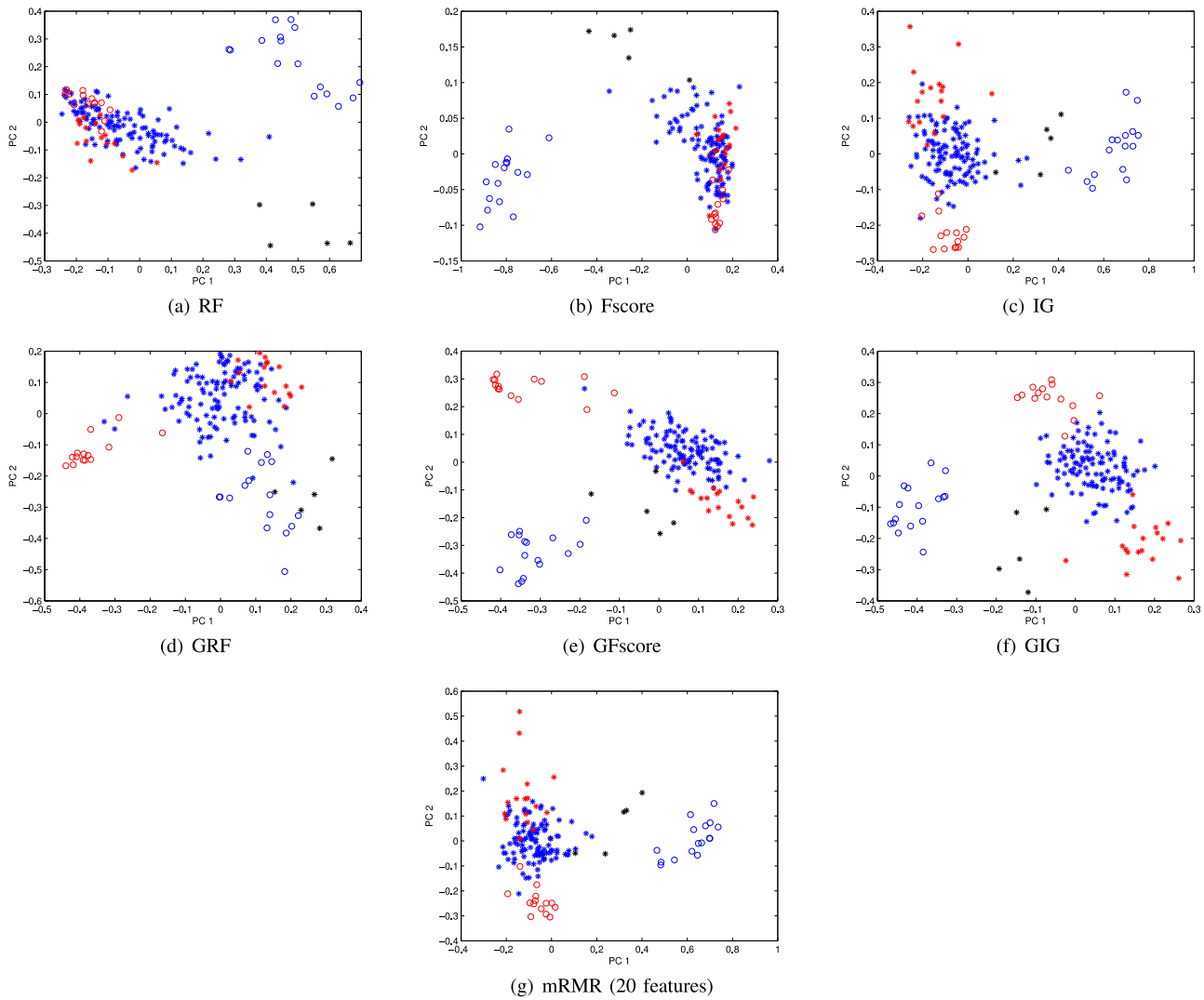


Fig. 6. Projection on first two principle components using top 20 features selected by different supervised feature selection algorithms on the LUNG data set.

Figs. 6 and 7, we can see that: for all three methods, the originally entangled three classes (red circle, red star, and blue star) became much more separable after applying the GRM framework. Projection of mRMR method separates data samples better than RF, Fscore, and IG. But after applying the GRM framework, the projection of GRF, GFscore, and GIG are more separable than mRMR. This justify the advantage of using global redundancy minimization over the greedy search approach adopted by mRMR.

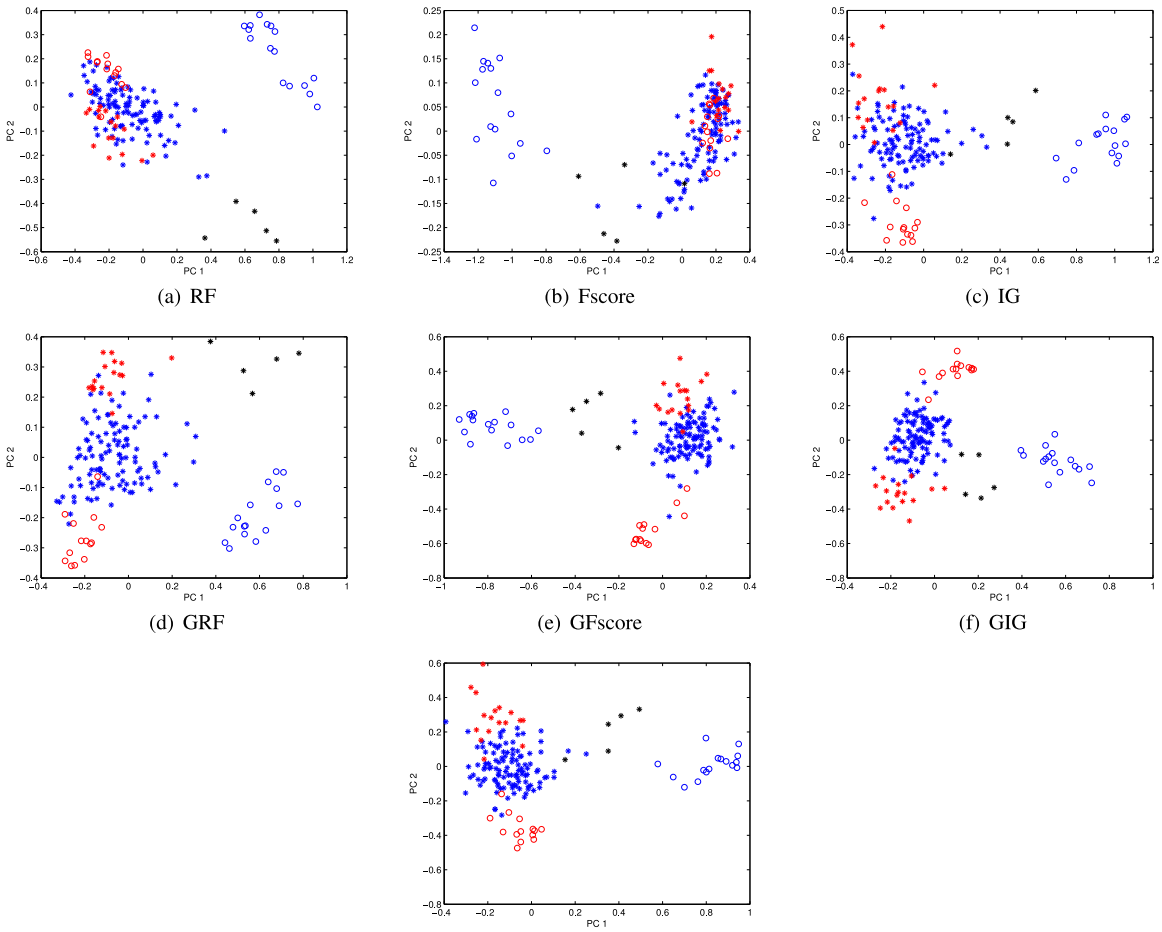
4.6 GRM Framework with Prior Group Knowledge Constraint

In many biological applications, it is desirable to select features (genes, biomarkers) in the same group that act together in the regulation pathway controlling a specific desease. So group constraint can be added into the GRM framework in order to select features in the same group. We first use the three unsupervised feature ranking criterion to select top 200 features. Then we apply the GRM framework to get the refined feature ranking of the top 200 features. After that, GRM framework with group constraint is performed in order to select features within the same group. To simulate the group constraint, we randomly select 100 features from the top 200 features, and then group these 100 features into 50 groups randomly, with two features in each group. We regard it as a *correct move* if the difference of rank of two features in the same group become smaller after applying the GRM framework with group constraint. Features in the group making a correct move would be more likely to be selected or excluded together, since the rank of features in this group became similar.

To measure whether the model tends to select features in the same group, we computed the correct move rate (the number of correct move feature groups divided by the number of total groups), as shown in Table 8. We can see that the correct move rate of using all three methods are pretty high. Averaged on six data sets, all the correct move rate exceeds 0.9. This justify that the GRM framework with group constraint tends to select features in the same group.

5 Conclusion

In this paper, we have proposed a framework for global redundancy minimization. The redundancy is reduced by



(a) RF (b) Fscore (c) IG

(d) GRF (e) GFscore (f) GIG

(g) mRMR (40 features)

Fig. 7. Projection on first two principle components using top 40 features selected by different supervised feature selection algorithms on the LUNG data set.

applying the GRM framework, and classification accuracy has improved significantly for both unsupervised and supervised feature selection algorithms. This demonstrate the effectiveness of the GRM framework, which minimize the redundancy between selected features,thus, the selected features are expected to be more compact and discriminant.

Acknowledgments

This research was partially supported by the US National Science Foundation (NSF)-IIS 1117965, NSF-IIS 1302675, NSF-IIS 1344152, NSF-DBI 1356628. Heng Huang is the corresponding author.

TABLE 8
Correct Move Rate of GRM Framework with Group Constraint

	Var	Lap	LGD
GLIOMA	0.908	0.732	0.872
LUNG	0.944	0.952	0.972
AMLALL	0.952	0.932	0.924
ProCancer	0.924	0.912	0.952
LEU	0.956	0.960	0.972
SRBCT	0.936	0.960	0.944
Average	0.937	0.908	0.939

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.

[2] P. N. Belhumeur, J. P. Hespanha, and D. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[3] D. P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Belmont, MA, USA: Athena Scientific, 1996.

[4] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2010, pp. 333–342.

[5] X. Cai, F. Nie, and H. Huang, "Exact top-k feature selection via l2,0-norm constraint," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 1240–1246.

[6] X. Cai, F. Nie, H. Huang, and C. Ding, "Feature selection via l2,1-norm support vector machine," in *Proc. IEEE Int. Conf. Data Mining*, 2011, pp. 91–100.

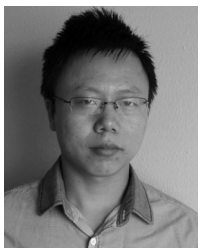
[7] X. Chang, F. Nie, Y. Yang, and H. Huang, "A convex formulation for semi-supervised multi-label feature selection," in *Proc. AAAI Conf. Artif. Intell.*, 2014, pp. 1171–1177.

[8] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," *J. Bioinformatics Comput. Biol.*, vol. 3, no. 02, pp. 185–205, 2005.

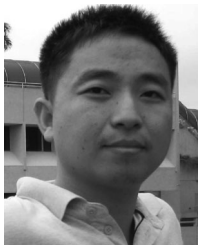
[9] M. Douze, H. Jégou, H. Sandhawalia, L. Amsaleg, and C. Schmid, "Evaluation of gist descriptors for web-scale image search," in *Proc. ACM Int. Conf. Image Video Retrieval*, 2009, p. 19.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. New York, NY, USA: Wiley, 2012.

[11] G. Forman and E. Kirshenbaum, "Extremely fast text feature extraction for classification and indexing," in *Proc. Int. Conf. Inf. Knowl. Manag.*, 2008, pp. 1221–1230.
[12] P. Gong, K. Gai, and C. Zhang, "Efficient euclidean projections via piecewise root finding and its application in gradient projection," *Neurocomputing*, vol. 74, no. 17, pp. 2754–2766, 2011.
[13] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learning Res.*, vol. 3, pp. 1157–1182, 2003.
[14] I. Guyon, S. R. Gunn, A. Ben-Hur, and G. Dror, "Result analysis of the nips 2003 feature selection challenge," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 545–552.
[15] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. Adv. Neural Inf. Process. Syst.*, 2006, vol. 18, p. 507.
[16] M. Hestenes, "Multiplier and gradient methods," *J. Optimization Theory Appl.*, vol. 4, no. 5, pp. 303–320, 1969.
[17] X. Jin, A. Xu, R. Bie, and P. Guo, "Machine learning techniques and chi-square feature selection for cancer classification using SAGE gene expression profiles," in *Proc. Int. Conf. Data Mining Biomedical Appl.*, 2006, pp. 106–115.
[18] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, "Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks," *Nature Med.*, vol. 7, no. 6, pp. 673–679, 2001.
[19] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artif. Intell.*, vol. 97, no. 1/2, pp. 273–324, 1997.
[20] I. Kononenko, "Estimating attributes: Analysis and extensions of relief," in *Proc. Eur. Conf. Mach. Learning*, 1994, pp. 171–182.
[21] D. D. Lewis, "Feature selection and feature extraction for text categorization," in *Proc. Workshop Speech Natural Language*, Stroudsburg, PA, USA, 1992, pp. 212–217.
[22] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," *arXiv preprint arXiv:1009.5055*, 2010.
[23] H. Liu and H. Motoda, *Feature Selection for Knowledge Discovery and Data Mining*. New York, NY, USA: Springer, 1998.
[24] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proc. 7th IEEE Int. Conf. Comput. Vis.*, 1999, vol. 2, pp. 1150–1157.
[25] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (coil-20)," Columbia University, Tech. Rep. CUCS-005-96, 1996.
[26] F. Nie, H. Huang, X. Cai, and C. Ding, "Efficient and robust feature selection via joint l2, 1-norms minimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2010, vol. 23, pp. 1813–1821.
[27] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1226–1238, Aug. 2005.
[28] M. J. D. Powell, "A method for nonlinear constraints in minimization problems," in *Optimization*, R. Fletcher, Ed. New York, NY, USA: Academic, 1969.
[29] L. E. Raileanu and K. Stoffel, "Theoretical comparison between the GINI index and information gain criteria," Univ. Neuchatel, Neuchatel, Switzerland, 2000.
[30] I. Rodriguez-Lujan, R. Huerta, C. Elkan, and C. S. Cruz, "Quadratic programming feature selection," *J. Mach. Learning Res.*, vol. 11, pp. 1491–1516, 2010.
[31] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.
[32] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *J. Roy. Statist. Soc B.*, vol. 58, pp. 267–288, 1996.
[33] D. Wang, F. Nie, and H. Huang, "Unsupervised feature selection via unified trace ratio formulation and k-means clustering (track)." in *Proc. Eur. Conf. Mach. Learning Knowl. Discovery Databases*, 2014, pp. 306–321.
[34] D. Wang, F. Nie, H. Huang, J. Yan, S. L. Risacher, A. J. Saykin, and L. Shen, "Structural brain network constrained neuroimaging marker identification for predicting cognitive functions," in *Proc. 23rd Int. Conf. Inf. Process. Med. Imaging*, 2013, pp. 536–547.
[35] D. Wang, L. Yang, Z. Fu, and J. Xia, "Prediction of thermophilic protein with pseudo amino acid composition: An approach from combined feature selection and reduction," *Protein Peptide Lett.*, vol. 18, no. 7, pp. 684–689, 2011.
[36] H. Wang, F. Nie, H. Huang, S. Kim, K. Nho, S. L. Risacher, A. J. Saykin, and L. Shen, "Identifying quantitative trait loci via group-sparse multitask regression and feature selection: An imaging genetics study of the ADNI cohort," *Bioinformatics*, vol. 28, no. 2, pp. 229–37, 2012.
[37] H. Wang, F. Nie, H. Huang, S. Risacher, C. Ding, A. J. Saykin, L. Shen, and ADNI, "Sparse multi-task regression and feature selection to identify brain imaging predictors for memory performance," in *Proc. IEEE Conf. Comput. Vis.*, 2011, pp. 557–562.
[38] X. Wang, T. X. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 32–39.
[39] J. Wu and J. M. Rehg, "CENTRIST: A visual descriptor for scene categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1489–1501, Aug. 2011.



De Wang is currently working toward the PhD degree at the University of Texas at Arlington under the supervision of Dr. Heng Huang. His research interests include machine learning, data mining, and computer vision. His publications appear in proceedings of prestigious international conference like SIGKDD, ECML/PKDD, IPMI, and MICCAI.



Feiping Nie received the PhD degree in computer science from Tsinghua University, China, in 2009. He is currently a research assistant professor at the University of Texas, Arlington. His research interests are machine learning and its application fields, such as pattern recognition, data mining, computer vision, image processing, and information retrieval. He has published more than 100 papers in the prestigious journals and conferences like *TPAMI*, *TKDE*, ICML, NIPS, KDD, etc. He is currently serving as an associate editor or a PC member for several prestigious journals and conferences in the related fields.

Heng Huang received the BS and MS degrees from Shanghai Jiao Tong University, Shanghai, China, in 1997 and 2001, respectively. He received the PhD degree in computer science from Dartmouth College in 2006. Since 2012, he has been an associate professor in the Computer Science and Engineering Department, University of Texas at Arlington. His research interests include machine learning, data mining, bioinformatics, neuroinformatics, and health informatics.