Knowledge-Based Systems 110 (2016) 267–274

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Global feature selection from microarray data using Lagrange multipliers

Shiquan Sun[a,b], Qinke Peng[a,*], Xiaokang Zhang[a]

[a] Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China
[b] Department of Biostatistics, University of Michigan, Ann Arbor, MI 48109, USA

ARTICLE INFO

Article history:
Received 19 December 2015
Revised 25 July 2016
Accepted 26 July 2016
Available online 27 July 2016

Keywords:
Gene selection
Max-cut
Semidefinite programming
Lagrange multiplier
High-dimensional and small sample size

ABSTRACT

In microarray-based gene expression analysis, thousands of genes are involved to monitor their expression levels under a particular condition. In fact, however, only few of them are highly expressed, which has been proven by Golub et al. How to identify these discriminative genes effectively is a significant challenge to risk assessment, diagnosis, prognostication in growing cancer incidence and mortality.

In this paper, we present a global feature selection method based on semidefinite programming model which is relaxed from the quadratic programming model with maximizing feature relevance and minimizing feature redundancy. The main advantage of relaxation is that the matrix in mathematical model only requires symmetric matrix rather than positive (or semi) definite matrix. In semidefinite programming model, each feature has one constraint condition to restrict the objective function of feature selection problem. Herein, another trick in this paper is that we utilize Lagrange multiplier as proxy measurement to identify the discriminative features instead of solving a feasible solution for the original max-cut problem. The proposed method is compared with several popular feature selection methods on seven microarray data sets. The results demonstrate that our method outperforms the others on most data sets, especially for the two hard feature selection data sets, Beast(Wang) and Medulloblastoma.

1. Introduction

In microarray data analysis, identifying the disease-associated genes has been proven to be an important way to a certain diagnosis, therapy, and cancer prognosis [8]. This type of data may easily come out with thousands of features[1]. However, many of them may be redundant or irrelevant to the prediction [32,33]. Moreover, the situation in which is likely to render the risk of overfitting and easy to increase the computational burden of processing [21,37]. Therefore, selecting a discriminative and parsimonious subset of features before performing classification is a very important task for the analysis of microarry gene expression data.

In the last decades, a considerable effort has been devoted to developing feature selection procedures. These methods designed with different criteria broadly fall into three categories, namely filter (classifier-independent) [26,38], wrapper (classifier-dependent) [2,24] and embedded [39] (classifier-dependent). Compared with the other two types of feature selection methods, the filter method, which is relatively cheap in terms of computational expense and easy to avoid over-fitting, is widely used for feature selection problem. In general, the filter method defines a score to act as a proxy measurement of the importance of a specific feature, particularly for iterative-based greedy methods such as Relief [29] and mRMR [32]. Brown et al. summarized seventeen mutual information or conditional mutual information-based feature selection methods and developed a unifying framework for these methods [6]. This work can guide us in various situations to choose a appropriate method in feature selection problem. However, these iterative methods are greedy in nature, and therefore are prone to obtain sub-optimal solutions in feature selection problem. For example, suppose we want to select two features from given features $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{x}_3$, and $\boldsymbol{x}_4$. The mutual information between features and class $\boldsymbol{y}$ are $I(\boldsymbol{x}_1;\boldsymbol{y}) = 0.56 I(\boldsymbol{x}_2;\boldsymbol{y}) = 0.33,\ I(\boldsymbol{x}_3;\boldsymbol{y}) = 0.25,$ and $I(\boldsymbol{x}_4;\boldsymbol{y}) = 0.43$. The conditional mutual information between features given class are $I(\boldsymbol{x}_1,\boldsymbol{x}_2|\boldsymbol{y}) = 0.62,\ I(\boldsymbol{x}_3,\boldsymbol{x}_4|\boldsymbol{y}) = 0.65,\ I(\boldsymbol{x}_1,\boldsymbol{x}_4|\boldsymbol{y}) = 0.09,$ $I(\boldsymbol{x}_2,\boldsymbol{x}_3|\boldsymbol{y}) = 0.19,\ I(\boldsymbol{x}_1,\boldsymbol{x}_3|\boldsymbol{y}) = 0.12,$ and $I(\boldsymbol{x}_2,\boldsymbol{x}_4|\boldsymbol{y}) = 0.05$. Actually, we will obtain a feature subset $\{\boldsymbol{x}_1, \boldsymbol{x}_2\}$ rather than the optimal a feature subset $\{\boldsymbol{x}_3, \boldsymbol{x}_4\}$ because dropping feature $\boldsymbol{x}_1$ will lose more information than the others.

To deal with this issue, theoretically, a promising way is to establish convex quadratic programming model, which can find a

* Corresponding author. Fax: +86-29-82667964.
E-mail addresses: shiquan_sun@126.com (S. Sun), qkpeng@xjtu.edu.cn, qkpeng@mail.xjtu.edu.cn (Q. Peng).

[1] we do not distinguish between gene and feature in this paper.

http://dx.doi.org/10.1016/j.knosys.2016.07.035
0950-7051/© 2016 Elsevier B.V. All rights reserved.

global optimal solution for feature selection problem [20]. However, the matrix $\boldsymbol{K}$ in quadratic model is usually indefinite (i.e. there exists the negative eigenvalue) in practical applications, which results in nonconvex optimization, especially in microarray data analysis. To ensure the positive(or semi) definiteness of matrix $\boldsymbol{K}$, an important work proposed by Rodriguez-Lujan et al. [36] (QPFS) is to use Nyström sampling method using low-rank approximation for $\boldsymbol{K}$. The drawback of QPFS presented by Nguyen et al. [31] is that self-redundancy should not be included in $\boldsymbol{K}$ which may result in selection bias. But if the self-redundancy term (the elements on the main diagonal of matrix $\boldsymbol{K}$) is omitted from the matrix $\boldsymbol{K}$, it will violate the positive definiteness.

An alternative way to deal with this problem is to relax the quadratic programming model as semi-definite programming model in which the matrix $\boldsymbol{K}$ only requires to ensure the symmetry. In this paper, instead of adopting a direct solving quadratic programming, we consider its semidefinite relaxation model in which each feature is restricted by one constraint condition. Further, there is no need to obtain the solution of primal problem via rounding method in the proposed approach. We just utilize the solution of dual problem, *Lagrange multiplier*, as proxy measurement to select the discriminative features for classification. Experimental results show that new mathematical model based on Lagrange multipliers is a competitive and efficient filter-type feature selection method for classification.

The paper is organized as follows: Section 2 devotes to establish a semi-definite programming model for feature selection problem. Section 3 explains why the Lagrange multiplier can be interpreted as a score for the process of feature selection. Section 4 conducts several experiments to verify the proposed model. The discussion and conclusion can be found in Section 5.

2. The optimization model for feature selection problem

Given a feature set $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n)$ where $\boldsymbol{x}_i \in \mathbb{R}^m$, and response variable (or target class variable) $\boldsymbol{y}$ where $\boldsymbol{y} \in \mathbb{R}^m$ (m is the number of samples and n is the number of features). Essentially, most of filtering feature selection methods share the same objective function, $\min_{\boldsymbol{w}} \mathcal{J}(\boldsymbol{w}) = \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{w}\|_2^2$. The goal is to find the optimal coefficient $\boldsymbol{w}$ as the score to select the discriminative features. Actually, we reformulate the equivalent form of the above optimization problem as $\min_{\boldsymbol{w}} \mathcal{J}(\boldsymbol{w}) = \boldsymbol{w}\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{w} - \beta\boldsymbol{y}^T\boldsymbol{X}\boldsymbol{w}$. To capture the nonlinear relationship between features, and features and the response variable $\boldsymbol{y}$, kernel functions are commonly considered in practical applications. Therefore, the above problem can be reformulated as

$$\min_{\boldsymbol{w}} \mathcal{J}(\boldsymbol{w}) = \boldsymbol{w}^T\boldsymbol{K}^{xx}\boldsymbol{w} - \beta\boldsymbol{K}^{xy}\boldsymbol{w} \tag{1}$$

where $\boldsymbol{K}^{xx}$ is the kernel matrix of data $\boldsymbol{X}$. That is,

$$\begin{pmatrix} \kappa_{11}^{xx} & \kappa_{12}^{xx} & \cdots & \kappa_{1n}^{xx} \\ \kappa_{21}^{xx} & \kappa_{22}^{xx} & \cdots & \kappa_{2n}^{xx} \\ \vdots & \vdots & \ddots & \vdots \\ \kappa_{n1}^{xx} & \kappa_{n2}^{xx} & \cdots & \kappa_{nn}^{xx} \end{pmatrix} = \boldsymbol{K}^{xx}.$$

and $\boldsymbol{K}^{xy}$ is the kernel vector between response variable $\boldsymbol{y}$ and features

$$\begin{pmatrix} \kappa_1^{xy} \\ \kappa_2^{xy} \\ \vdots \\ \kappa_n^{xy} \end{pmatrix} = \boldsymbol{K}^{xy}.$$

The first term in Eq. (1) is considered as a *redundancy* term and the second term is a *relevancy* term. This is the well-known relevancy-redundancy criterion in feature selection problem. In the following we will illustrate two families of feature selection problem which can fit this filter criterion, information theory-based feature selection methods which are belong to the iterative search paradigm, and quadratic programming-based feature selection methods which are the global search techniques.

2.1. Information theory-based feature selection methods

Information theoretic feature selection methods consider the elements of kernel matrix $\boldsymbol{K}^{xx}$ and $\boldsymbol{K}^{xy}$ in Eq. (1) as mutual information $I(\boldsymbol{x}_i;\ \boldsymbol{x}_j)$ or conditional mutual information $I(\boldsymbol{x}_i;\ \boldsymbol{x}_j|\boldsymbol{y})$. For this paradigm, there are two important papers [6,46]: Brown et al. summarized seventeen information theory-based filter methods over the last two decades and presented a unifying framework for these methods; and Vergara and Estvez presented a number of open problems in this field.

The generalized *minimum Redundancy Maximum Relevance* (mRMR) family framework [32] is parameterized as

$$\min_{\boldsymbol{x}_i} \mathcal{J}_{mrmr} = \sum_{\boldsymbol{x}_j \in \mathbb{S}} I\left(\boldsymbol{x}_i; \boldsymbol{x}_j\right) - \beta I(\boldsymbol{x}_i; \boldsymbol{y}) \tag{2}$$

where $\mathbb{S}$ is the selected feature subset already. Alternatively, we can replace $I(\boldsymbol{x}_i;\ \boldsymbol{x}_j)$ with $I(\boldsymbol{x}_i;\ \boldsymbol{x}_j|\boldsymbol{y})$ or set the coefficient β with different values, it will fall into different feature filter methods. For example, if we replace $I(\boldsymbol{x}_i;\ \boldsymbol{x}_j)$ with $I\left(\boldsymbol{x}_i; \boldsymbol{x}_j\right) - I\left(\boldsymbol{x}_i; \boldsymbol{x}_j|\boldsymbol{y}\right)$ and set β as the cardinality of selected feature subset $\mathbb{S}$, it becomes the well-known filter method *Joint Mutual Information* (JMI), i.e.,

$$\min_{\boldsymbol{x}_i} \mathcal{J}_{jmi} = \sum_{\boldsymbol{x}_j \in \mathbb{S}} \left\{ I\left(\boldsymbol{x}_i; \boldsymbol{x}_j\right) - I\left(\boldsymbol{x}_i; \boldsymbol{x}_j|\boldsymbol{y}\right) \right\} - |\mathbb{S}| I(\boldsymbol{x}_i; \boldsymbol{y}) \tag{3}$$

The QPFS proposed by Rodriguez-Lujan et al. [36] set the elements of kernel matrix $\boldsymbol{K}^{xx}$ as $\kappa_{ij}^{xx} = I\left(\boldsymbol{x}_i; \boldsymbol{x}_j\right)$ and $\kappa_{ii}^{xx} = I(\boldsymbol{x}_i; \boldsymbol{x}_i)$. Generally, the self-redundancy term κ_{ii}^{xx} might result in selection bias. However, if the κ_{ii}^{xx} is set as 0, the matrix $\boldsymbol{K}$ is indefinite, the optimal solution might not be found because of nonconvex objective function. However, this issue cannot affect our model. Therefore, we set $\kappa_{ij}^{xx} = I\left(\boldsymbol{x}_i; \boldsymbol{x}_j\right)$ and $\kappa_{ii}^{xx} = 0$.

2.2. Quadratic programming-based feature selection methods

Acutally, we can easily cast the Eq. (2) as quadratic programming feature selection (QPFS) problem if we fix the number of selected features,

$$\begin{aligned} \underset{\boldsymbol{w}}{\text{minimize}} \quad & \boldsymbol{w}^T\boldsymbol{K}^{xx}\boldsymbol{w} - \beta\boldsymbol{K}^{xy}\boldsymbol{w} \\ \text{subject to} \quad & \textstyle\sum_{i=1}^n w_i = k, \\ & w_i \in \{0, 1\}, i = 1, 2, \cdots, n. \end{aligned} \tag{4}$$

where $\boldsymbol{K}_{n\times n}^{xx}$ is a matrix of feature pairwise redundancy. $\boldsymbol{K}_{n\times 1}^{xy}$ is a vector of feature relevancy. Unfortunately, it is an NP-hard problem. An alternative way is to drop the 0–1 integer programming problem into continuous optimization problem resulting in,

$$\begin{aligned} \underset{\boldsymbol{w}}{\text{minimize}} \quad & \boldsymbol{w}^T\boldsymbol{K}^{xx}\boldsymbol{w} - \beta\boldsymbol{K}^{xy}\boldsymbol{w} \\ \text{subject to} \quad & \textstyle\sum_{i=1}^n w_i = k, \\ & w_i \geq 0, i = 1, 2, \cdots, n. \end{aligned} \tag{5}$$

The most attractive characteristic of the QPFS is that it can obtain the globally optimal solution if the matrix $\boldsymbol{K}^{xx}$ is positive (or semi) definiteness. In practice, it is fail to guarantee the positive definiteness, especially in high-dimensional and small sample size data sets.

To decrease the condition of positive (or semi) definiteness, we continue to relax the optimization model as semi-definite programming (SDP) model. Based on Eq. (4), we can reformulate $\boldsymbol{K} =$

$\boldsymbol{K}^{xx} - \beta diag(\boldsymbol{K}^{xy})$ because of the variable $w_i \in \{0, 1\}$, therefore, the Eq. (4) can be rewritten as follows without considering the fixed number of selected features.

$$\min_{\boldsymbol{w} \in \{0,1\}^n} \mathcal{J}(\boldsymbol{w}) = \boldsymbol{w}^T \boldsymbol{K} \boldsymbol{w}$$

We consider the semidefinite relaxation of quadratic 0–1 optimization. This problem is equivalent to max-cut problem [25]. Therefore, we use simple variable transformation $w_i = \frac{z_i+1}{2}$ to convert variable $w_i \in \{0, 1\}$ into the variable $z_i \in \{-1, 1\}$ in max-cut problem. Now the new objective function becomes

$$\min_{\boldsymbol{z} \in \{-1,1\}^n} \mathcal{J}(\boldsymbol{z}) = (\boldsymbol{z} + \boldsymbol{e})^T \boldsymbol{K} (\boldsymbol{z} + \boldsymbol{e})$$

where $\boldsymbol{e} \in \mathbb{R}^n$ is a column vector of all 1s.

A variable expansion trick can be applied to put both the transformed objective function and the constraint back into a nice quadratic form. Let $\boldsymbol{z} = (z_0, z_1, z_2, \cdots, z_n)$ and $z_0 = 1$. Then we can obtain a new form in term of the following matrix

$$\boldsymbol{C} = \begin{pmatrix} \boldsymbol{e}^T \boldsymbol{K} \boldsymbol{e} & \boldsymbol{e}^T \boldsymbol{K} \\ \boldsymbol{K} \boldsymbol{e} & \boldsymbol{K} \end{pmatrix}_{(n+1) \times (n+1)} \tag{6}$$

After this transformation, the Eq. (2) becomes the following optimization model

$$\begin{aligned} \underset{\boldsymbol{z}}{\text{minimize}} \quad & \boldsymbol{z}^T \boldsymbol{C} \boldsymbol{z} \\ \text{subject to} \quad & z_0 = 1, \\ & z_i \in \{-1, 1\}, i = 1, 2, \cdots, n. \end{aligned} \tag{7}$$

We now return our attention to the semidefinite relaxation for the model (7). Let $\boldsymbol{Z} = \boldsymbol{z}\boldsymbol{z}^T$, the above problem is equivalent to the following (convex) SDP in primal form by dropping the rank constraint $rank(\boldsymbol{Z}) = 1$.

$$\begin{aligned} \underset{\boldsymbol{Z}}{\text{minimize}} \quad & \text{trace}(\boldsymbol{C}\boldsymbol{Z}) \\ \text{subject to} \quad & \text{trace}(\boldsymbol{e}_i \boldsymbol{e}_i^T \boldsymbol{Z}) = 1, i = 0, 1, \cdots, n, \\ & \boldsymbol{Z} \succeq 0. \end{aligned} \tag{8}$$

Here $\boldsymbol{Z} \succeq 0$ represents that the matrix $\boldsymbol{Z}$ is a positive semidefinite matrix, and trace$(\cdot)$ is the trace of matrix. The vector $\boldsymbol{e}_i \in \mathbb{R}^{n+1}$ represents a unit column vector that the $(i+1)$-th element is equal to one.

We also consider the SDP problem (8) in a dual form. Applying the Lagrange multiplier technique we obtain the Lagrange function

$$\mathcal{L}(\boldsymbol{Z}, \boldsymbol{\lambda}, \boldsymbol{S}) = \text{trace}(\boldsymbol{C}\boldsymbol{Z}) - \sum_{i=0}^{n} \lambda_i (\text{trace}(\boldsymbol{e}_i \boldsymbol{e}_i^T \boldsymbol{Z}) - 1) - \text{trace}(\boldsymbol{S}\boldsymbol{Z}).$$

For $\forall \boldsymbol{\lambda} = \begin{pmatrix} \lambda_0 \\ \vdots \\ \lambda_n \end{pmatrix} \in \mathbb{R}^{n+1}$, $\forall \boldsymbol{S} \succeq 0$, we take the partial derivative of $\mathcal{L}$ with respect to the primal variable $\boldsymbol{Z}$ and then set this partial derivative equal to zero, namely

$$\frac{\partial \mathcal{L}(\boldsymbol{Z}, \boldsymbol{\lambda}, \boldsymbol{S})}{\partial \boldsymbol{Z}} = \boldsymbol{C} - \sum_{i=0}^{n} \lambda_i \left(\boldsymbol{e}_i \boldsymbol{e}_i^T\right) - \boldsymbol{S} = 0.$$

Then, substituting the above equation into $\mathcal{L}\big(\boldsymbol{Z}, \boldsymbol{\lambda}, \boldsymbol{S}\big)$, we have the following dual objective function

$$\begin{aligned} \underset{\boldsymbol{\lambda}, \boldsymbol{S}}{\text{minimize}} \quad & \boldsymbol{e}^T \boldsymbol{\lambda} \\ \text{subject to} \quad & \sum_{i=0}^{n} \lambda_i (\boldsymbol{e}_i \boldsymbol{e}_i^T) + \boldsymbol{S} \quad = \boldsymbol{C}, \\ & \boldsymbol{\lambda} \in \mathbb{R}^{n+1}, \boldsymbol{S} \succeq 0. \end{aligned} \tag{9}$$

Here $(\boldsymbol{\lambda}, \boldsymbol{S})$ is a feasible solution for the dual problem and the $\boldsymbol{\lambda}$ is the *Lagrange Multiplier*.

We solve this problem using *infeasible path* − *following* algorithm, which solves the pair of SDP Eqs. (8) and (9) simultaneously, and the package, SDPT-3 implemented in MATLAB, is freely available software (This package is at: http://www.math.nus.edu.sg/~mattohkc/sdpt3.html). Once the optimum solution $(\boldsymbol{\lambda}^*, \boldsymbol{S}^*)$ of dual problem is obtained, we do not need to generate a feasible solution to an original discrete problem via rounding method. We only use the optimal Lagrange multiplier $\boldsymbol{\lambda}^*$ to act as a proxy measurement of the importance of features. The next section will answer the question that why the Lagrange multiplier can be interpreted as the score of features.

To clearly represent LM method in feature selection process, we summarize the procedure of LM in Algorithm 1. Suppose we want to select a feature subset $\mathbb{S}$ with size k.

Algorithm 1 Feature selection using Lagrange multipliers (LM)

```
Data: $\boldsymbol{X}, \boldsymbol{y}$
Result: Selected gene subset $\mathbb{S}$
Normalize the data $\boldsymbol{X}$
Calculate the matrix $\boldsymbol{C}$ in Equation (6)
repeat
    Solve the primal problem (Equation (8)), $\boldsymbol{z}$
    Solve the dual problem (Equation (9)), $\boldsymbol{\lambda}$
until $\|\boldsymbol{z} - \boldsymbol{\lambda}_{-1}\| \le \epsilon$
According to the rank of the optimal Lagrange multiplier $\boldsymbol{\lambda}^*$, select top $k$ genes as $\mathbb{S}$
Return: $\mathbb{S}$
```

3. The interpretation of Lagrange multiplier for feature selection

This section is devoted to interpreting how Lagrange multipliers enable us to select features. Intuitively, in real applications, the constraint function can be thought of as "*competing*" with the desired objective function to "*pull*" the solution to its minimum (or maximum). The Lagrange multiplier can be thought of as a measure of how hard it is to pull in order to make those "*forces*" balance out on the constraint surface. This can be naturally generalized to multiple constraints, which typically "*pull*" in different directions. Therefore, the value of Lagrange multipliers can be interpreted as the magnitude of force in those directions. In the following we illustrate two applications for our inspiration regardless of their theoretical aspects.

The first inspiration we use Lagrange multipliers to select features is from support vector machine (SVM) which is quadratic programming model described in [45]. It is well known that the mathematical model of SVM is established in feature space. Each sample has one constraint, and the total number of constraints is m.

$$\begin{aligned} \underset{\boldsymbol{w}}{\text{minimize}} \quad & \frac{1}{2} \boldsymbol{w}^T \boldsymbol{w} \\ \text{subject to} \quad & y_i \left(\boldsymbol{w}^T x_i + b\right) \ge 1, i = 1, 2, \cdots, m. \end{aligned} \tag{10}$$

In general, instead of solving quadratic programming model, it is usually much easier to deal with its dual form formulating as follows.

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize}} \quad & \mathcal{J}(\boldsymbol{\alpha}) = \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle - \sum_{i=1}^{m} \alpha_i \\ \text{subject to} \quad & \sum_{i=1}^{m} \alpha_i y_i = 0, \alpha_i \ge 0, i = 1, 2, \cdots, m. \end{aligned} \tag{11}$$

The solution of dual form $\boldsymbol{\alpha}$, i.e. Lagrange multiplier, commonly indicates whether the sample contributes to the classification model or not. In fact, only a few elements of $\boldsymbol{\alpha}$ are nonzero

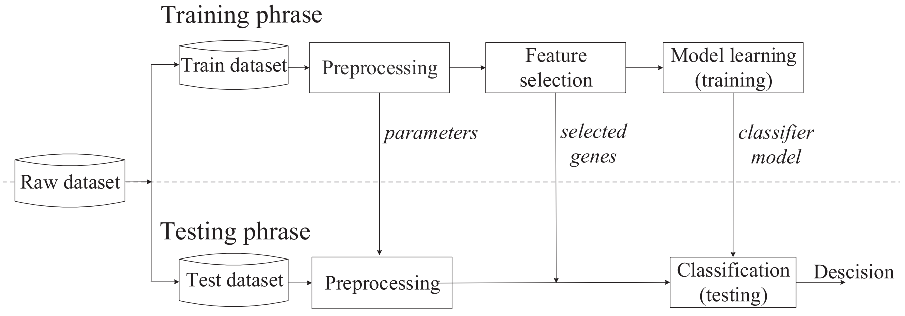

Fig. 1. The experimental diagram in our experiment. It is mainly divided into three parts: preprocessing phrase, feature selection phrase, and model training and prediction phrase.

(denoted as *nSV*, $nSV < m$) and their corresponding samples are so-called "*support vectors*". In our case, we try to establish the optimization model in sample space, and construct each constraint for each feature (see optimization model (8)), the solution of its dual form indicates the magnitude of each feature to contribute the objective function in feature selection problem (see optimization model (9)). Note that, one popular SVM-based iterative method is SVMrfe proposed by Rakotomamonjy [35] and is widely used in feature selection problem.

The second inspiration we use Lagrange multipliers to select features is from mathematical optimization. Lagrange multiplier method is a powerful tool for solving the optimization problems with implicit constraints and eliminating extra variables. From the sensitivity theorem [3], we notice that Lagrange multiplier can be interpreted as *the rate of change of the optimal cost as the level of constraints changes*, i.e.,

$$\mathcal{F}(\boldsymbol{\lambda}, \boldsymbol{x}) = \|\nabla\mathcal{U}(\boldsymbol{x}) - \sum_{i=1}^{n} \lambda_i \nabla\mathcal{D}_i(\boldsymbol{x})\|_2^2.$$

where $\mathcal{U}$ is a cost function and $\mathcal{D}_i$ is a constraint function. $\mathcal{F}(\boldsymbol{\lambda}, \boldsymbol{x}) = 0$ when the $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ achieve the optimal value $\boldsymbol{x}^*$ and $\boldsymbol{\lambda}^*$, respectively. We compare with the objective function of feature selection, which can be re-expressed as follows,

$$\mathcal{J}(\boldsymbol{w}) = \|\boldsymbol{y} - \sum_{i=1}^{n} w_i \boldsymbol{x}_i\|_2^2.$$

where the $\boldsymbol{w}$ is regression coefficient and $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n)$. Actually, with respect to $\mathcal{F}(\boldsymbol{\lambda}, \boldsymbol{x})$, it has been widely used for the identification of force and potential energy in physics [16], the decision-making analysis in economics [12] (Lagrange multiplier usually interpreted as the "*shadow price*" of its constraints), topology error identification in engineering [9] and others [22].

Inspired by the two examples mentioned above, therefore, it is very straightforward to switch our attention to selecting features using Lagrange multipliers. The magnitudes of Lagrange multipliers indicate the degree of importance of features, in other words, the contribution of features to the objective function.

4. Experiments

In this section, we shall empirically provide insights into the performance of proposed method (LM) and other six feature selection methods including five iterative feature selection methods, Relief, FBCF, mRMR, JMI, and SVMrfe; and a global feature selection method, QPFS. All experiments are conducted with five-fold cross validation (CV) which means each data set is randomly partitioned into 5 parts, four parts are used as training set, and the remaining one is used as testing set.

4.1. Experimental procedure

The experimental procedure is illustrated diagrammatically in Fig. 1. The first step is splitting the raw data set into two parts (i.e. training data set and test data set). In training phase, the feature selection module is to select the discriminative features from whole feature set, and then a learning model is trained using selected features. In testing phase, the trained model makes decision automatically for test data set. In preprocessing module of our case, the data $\boldsymbol{X}$ is centered and normalized such that each feature has zero-mean and one-standard deviation. Note that, the training data set and the test data set must be handled separately and the test data set is normalized using mean and variance which are from training data set.

Table 1
Microarray gene expression data sets used in our experiments.

Dataset ID	Dataset name	Features	Samples	P/N	References
MT1	AMLALL	7129	72	38/34	[17]
MT2	Breast cancer	22,283	209	138/71	[47]
MT3	Colon	2000	62	40/22	[1]
MT4	DLBCL	7129	77	58/19	[40]
MT5	Lung	7129	86	62/24	[18]
MT6	Medulloblastoma	7129	60	39/21	[34]
MT7	Prostate cancer	12,600	102	52/50	[41]

In this paper, we focus on the comparison of feature selection methods, while classification methods are used to train the model using the features which are selected by those methods. Over the past few years, many different kinds of feature selection methods have been developed. A well-known example is Relief [23], which is to update the score (or weight) of each feature according to its ability to discriminate samples with different class. However, Relief may fail to remove the features that are highly correlated with the discriminative features. In other words, it cannot identify redundant features. Generally, redundant features should be removed as well because they also affect the accuracy of prediction and the speed of training classification models. Fast Correlation Based Feature selection method (FCBF) designed by [28] is a typical method to separate relevant and redundant features based on pairwise correlation (i.e. Symmetrical Uncertainty). However, FCBF does not identify redundant features precisely in practice [7,42]. The other two popular methods are Minimum Redundancy and Maximum Relevance (mRMR) [32] and Joint Mutual Information (JMI) [48]. JMI contains the conditional redundancy term while mRMR criterion omits the conditional redundancy term (Section 2.1). In practice, JMI outperforms mRMR in some cases while mRMR perform well than JMI in other cases [6]. Another important feature selection method is Support Vector Machine recursive feature elimination (SVMrfe) [19], which is a quadratic programming-based feature selection method. However, in principal, the methods mentioned above are iterative feature selection methods. A potential drawback is that once a feature is selected, it cannot be dropped at later stage (a simple example in Introduction). Therefore, to avoid obtaining suboptimal solution for

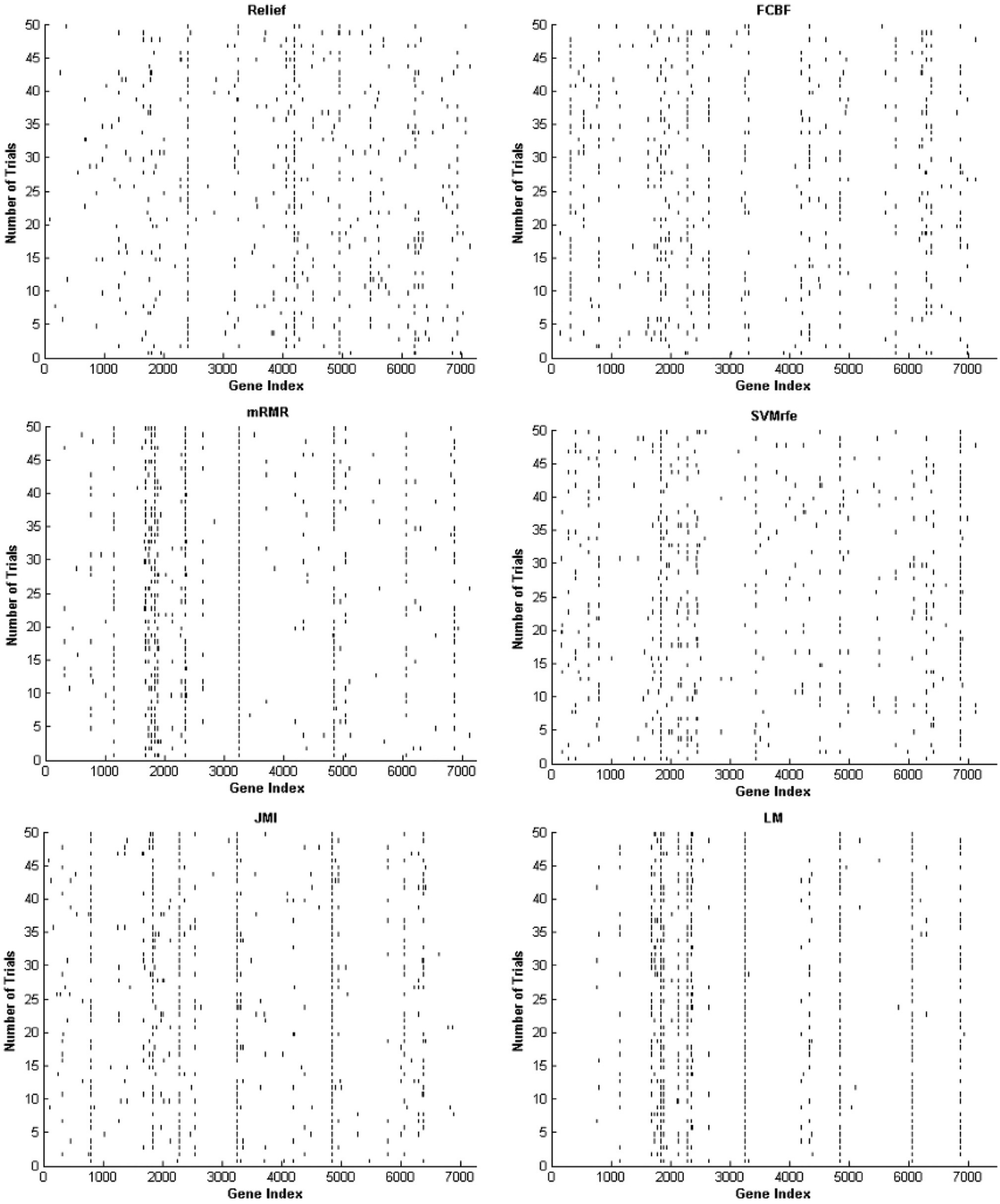

Fig. 2. Visualizing stability performance of six feature selection procedures on MT1 data set with top 10 features. A point indicates that the corresponding feature is selected per trial. The more complete vertical lines the algorithm has (i.e., same feature selected among different trials), the more stable it is.

feature selection problem, a Quadratic Programming Feature Selection method (QPFS) [36] was proposed to find the global solution for feature selection problem. However, several non-trivial issues are restricted to its applications, such as requiring the matrix $\boldsymbol{K}$ is positive definition. Therefore, in this paper, we relax the quadratic programming to semi-definite programming for feature selection problem.

4.2. Microarray gene expression data sets

Microarray gene expression-based cancer classification is one of the most important tasks to a certain cancer prognosis. A typical

Table 2
Comparison of feature selection methods on seven microarray data sets.

Selector	Naive Bayes		KNN		CART		Random forest	
	Acc	AUC	Acc	AUC	Acc	AUC	Acc	AUC
					MT1			
Relief	0.923(7)	0.955(7)	0.883(7)	0.946(7)	0.91(7)	0.961(6)	0.944(7)	0.955(6)
FCBF	0.958(6)	0.974(6)	0.944(6)	0.953(6)	0.925(6)	0.954(7)	0.956(6)	0.949(7)
mRMR	0.961(4.5)	0.982(4)	0.953(4)	0.968(5)	0.934(5)	0.968(5)	0.958(5)	0.958(5)
SVMrfe	0.965(3)	0.983(3)	0.962(3)	0.973(2)	0.94(2.5)	0.975(3)	0.966(3)	0.962(4)
JMI	0.961(4.5)	0.977(5)	0.951(5)	0.969(4)	0.938(4)	0.972(4)	0.961(4)	0.968(3)
QPFS	0.966(2)	0.985(2)	0.964(2)	0.971(3)	0.94(2.5)	0.976(2)	0.968(2)	0.97(2)
LM	0.968(1)	0.987(1)	0.974(1)	0.975(1)	0.941(1)	0.979(1)	0.979(1)	0.988(1)
					MT2			
Relief	0.646(7)	0.689(7)	0.665(7)	0.692(7)	0.613(7)	0.679(7)	0.681(7)	0.725(7)
FCBF	0.725(6)	0.764(6)	0.713(6)	0.759(5)	0.677(6)	0.73(6)	0.747(6)	0.795(6)
mRMR	0.771(3)	0.784(4)	0.749(4)	0.784(3)	0.719(2)	0.806(2)	0.781(2)	0.807(3)
SVMrfe	0.739(5)	0.772(5)	0.745(5)	0.757(6)	0.698(5)	0.798(4)	0.761(4.5)	0.802(5)
JMI	0.748(4)	0.786(3)	0.755(2)	0.786(2)	0.701(4)	0.788(5)	0.761(4.5)	0.806(4)
QPFS	0.792(1)	0.806(2)	0.751(3)	0.76(4)	0.704(3)	0.801(3)	0.763(3)	0.814(2)
LM	0.786(2)	0.817(1)	0.756(1)	0.791(1)	0.723(1)	0.816(1)	0.788(1)	0.817(1)
					MT3			
Relief	0.815(7)	0.823(7)	0.856(7)	0.872(6)	0.797(6)	0.839(6)	0.832(7)	0.86(7)
FCBF	0.827(6)	0.835(6)	0.857(6)	0.865(7)	0.789(7)	0.832(7)	0.851(6)	0.864(6)
mRMR	0.859(5)	0.871(4)	0.874(2.5)	0.878(3)	0.83(2)	0.858(2)	0.869(3)	0.882(4)
SVMrfe	0.875(1.5)	0.878(3)	0.874(2.5)	0.886(2)	0.822(4)	0.855(3)	0.878(1)	0.891(2)
JMI	0.873(3)	0.889(1)	0.868(4)	0.877(4)	0.823(3)	0.848(4)	0.868(4)	0.883(3)
QPFS	0.869(4)	0.864(5)	0.859(5)	0.874(5)	0.809(5)	0.844(5)	0.857(5)	0.878(5)
LM	0.875(1.5)	0.887(2)	0.876(1)	0.896(1)	0.833(1)	0.879(1)	0.874(2)	0.892(1)
					MT4			
Relief	0.903(7)	0.928(7)	0.904(7)	0.931(6)	0.891(7)	0.938(7)	0.917(7)	0.971(6)
FCBF	0.931(5)	0.939(6)	0.908(6)	0.916(7)	0.903(6)	0.98(6)	0.921(6)	0.969(7)
mRMR	0.948(1)	0.956(3)	0.948(2)	0.977(3)	0.939(2)	0.985(3.5)	0.934(2)	0.989(2)
SVMrfe	0.935(4)	0.953(4)	0.928(4)	0.94(5)	0.923(4)	0.985(3.5)	0.929(4)	0.98(4)
JMI	0.939(3)	0.964(2)	0.94(3)	0.995(2)	0.929(3)	0.992(2)	0.93(3)	0.986(3)
QPFS	0.915(6)	0.943(5)	0.917(5)	0.961(4)	0.908(5)	0.981(5)	0.923(5)	0.972(5)
LM	0.946(2)	0.966(1)	0.958(1)	0.998(1)	0.946(1)	0.994(1)	0.94(1)	0.992(1)
					MT5			
Relief	0.718(7)	0.76(7)	0.744(7)	0.779(6)	0.731(7)	0.808(7)	0.738(6)	0.786(6)
FCBF	0.77(6)	0.795(6)	0.777(6)	0.726(7)	0.751(6)	0.84(5)	0.73(7)	0.775(7)
mRMR	0.834(3)	0.883(2)	0.803(1)	0.87(2)	0.835(3)	0.869(3)	0.846(3)	0.892(1.5)
SVMrfe	0.831(4)	0.865(4)	0.791(4)	0.853(4)	0.793(5)	0.839(6)	0.814(5)	0.851(5)
JMI	0.848(1)	0.878(3)	0.797(3)	0.859(3)	0.84(2)	0.877(2)	0.854(2)	0.89(3)
QPFS	0.796(5)	0.837(5)	0.787(5)	0.828(5)	0.802(4)	0.841(4)	0.826(4)	0.864(4)
LM	0.839(2)	0.885(1)	0.798(2)	0.872(1)	0.845(1)	0.885(1)	0.857(1)	0.892(1.5)
					MT6			
Relief	0.772(6)	0.816(6)	0.732(6)	0.753(7)	0.657(7)	0.771(6)	0.787(7)	0.793(7)
FCBF	0.763(7)	0.794(7)	0.722(7)	0.777(6)	0.697(6)	0.716(7)	0.817(6)	0.827(6)
mRMR	0.788(4)	0.83(3)	0.8(4)	0.818(4)	0.703(5)	0.773(5)	0.822(5)	0.839(5)
SVMrfe	0.807(3)	0.829(4)	0.815(3)	0.829(3)	0.768(1)	0.781(3)	0.847(1)	0.862(2)
JMI	0.785(5)	0.818(5)	0.792(5)	0.813(5)	0.712(4)	0.776(4)	0.842(3)	0.855(4)
QPFS	0.814(2)	0.837(1.5)	0.823(2)	0.841(2)	0.733(3)	0.784(2)	0.832(4)	0.857(3)
LM	0.817(1)	0.837(1.5)	0.825(1)	0.849(1)	0.743(2)	0.786(1)	0.846(2)	0.873(1)
					MT7			
Relief	0.913(6)	0.928(6)	0.888(7)	0.968(6)	0.84(7)	0.948(7)	0.922(7)	0.93(7)
FCBF	0.904(7)	0.914(7)	0.942(3)	0.978(2)	0.871(5)	0.949(6)	0.936(5)	0.942(6)
mRMR	0.942(2)	0.957(2)	0.945(2)	0.973(3)	0.875(2.5)	0.967(2)	0.945(2)	0.976(2)
SVMrfe	0.924(4)	0.95(3)	0.923(6)	0.967(7)	0.875(2.5)	0.956(3)	0.935(6)	0.973(4)
JMI	0.928(3)	0.937(4)	0.938(4.5)	0.972(4)	0.872(4)	0.952(4.5)	0.938(4)	0.974(3)
QPFS	0.92(5)	0.932(5)	0.938(4.5)	0.97(5)	0.868(6)	0.952(4.5)	0.943(3)	0.952(5)
LM	0.945(1)	0.968(1)	0.946(1)	0.986(1)	0.881(1)	0.968(1)	0.947(1)	0.978(1)

characteristic of this type of data is high-dimensional and small sample size. Because cancers are usually marked by changing in the expression levels of certain genes, therefore it is obvious that not all measured genes are discriminative genes. More importantly, the situation is likely to render the risk of overfitting and easy to increase the computational burden of processing. These pose great challenges to constructing an efficient classifier for prediction. Hence, feature selection(or gene selection) problem is ubiquitous in cancer classification.

To assess the performance of the proposed method, we conduct the experiments on a number of real microarray gene expression data sets which are described in detailed in Table 1. We have collected these data sets in our previous work [43] and they are freely available at https://github.com/sqsun/kernelPLS-datasets.

4.3. Results and analysis

In this section, we conduct the experiments on seven microarray gene expression data sets (Table 1) to show the effectiveness of the proposed method (LM). In order to show the feature selection methods which are classifier-independent in evaluation process, we use four classification methods, including CART (minparent=15), Naive Bayes (”kernel” distribution), KNN(k=5) and Random Forest (ntree=100), to train the classification model (Fig. 1). To obtain statistically reliable results, two evaluation cri-

Table 3
Friedman test for the comparison of LM with other feature selection methods with respect to each criterion (Acc or AUC).

	Naive Bayes		KNN		CART		Random forest	
	z	*p*-value	z	*p*-value	z	*p*-value	z	*p*-value
					Acc			
Relief.vs.LM	4.516	6.31E−06	4.949	7.47E−07	4.949	7.47E−07	4.825	1.40E−06
FCBF.vs.LM	4.021	5.80E−05	3.959	7.53E−05	4.206	2.59E−05	4.083	4.45E−05
mRMR.vs.LM	1.485	0.138	1.423	0.155	1.670	0.095	1.608	0.108
SVMrfe.vs.LM	1.732	0.083	2.412	0.016	1.979	0.048	1.918	0.055
JMI.vs.LM	1.608	0.108	2.289	0.022	1.979	0.048	1.918	0.055
QPFS.vs.LM	1.794	0.073	2.289	0.022	2.536	0.011	2.103	0.035
					AUC			
Relief.vs.LM	4.763	1.91E−06	4.701	2.59E−06	4.825	1.40E−06	4.763	1.91E−06
FCBF.vs.LM	4.392	1.12E−05	4.083	4.45E−05	4.578	4.70E−06	4.639	3.49E−06
mRMR.vs.LM	1.670	0.095	1.979	0.048	1.918	0.055	1.856	0.063
SVMrfe.vs.LM	2.165	0.030	2.722	0.006	2.289	0.022	2.289	0.022
JMI.vs.LM	1.794	0.073	2.103	0.035	2.289	0.022	1.918	0.055
QPFS.vs.LM	2.103	0.035	2.598	0.009	2.289	0.022	2.289	0.022

teria including Accuracy (Acc) and area under receiver operating characteristic curve (AUC) are used to measure the performance of each feature selection method. In training phrase, we take 50 trials in total.

The feature selection methods we compared in the experiments are, Relief, FCBF, JMI, mRMR, SVMrfe, and QPFS. For QPFS, we set parameter alpha as 0.75, the number of segments for discretization as 5, and the rate of sub-sampling in Nystörm method as 0.2. The parameters of the remaining methods are used with default settings.

The iterative feature selection methods strongly rely on the order of features which are incorporated. Therefore, we will show the stability of feature selection methods first. It is expected that the features selected by global feature selection method are more stable than iterative search methods. Without loss of generality, we use MT1 data set here. The experiment is conducted over 50 trials and top 10 features are considered for each method. The *x*-axis represents the feature index from 1 to 7129 (the total number of features is 7129) and the *y*-axis shows the 50 trials. For each trial, if the feature is selected, we plot a point at the corresponding position. The more complete vertical lines the method has, the more stable it is. As expected, from the Fig. 2, we can see that the features selected by LM are clearly the most stable method against the other methods. The second-best and third-best is mRMR and JMI, respectively. However, Relief, FCBF and SVMrfe show the poorer stability than the others. It should be noted that the good stability of methods does not mean to be high performance in evaluation criterion of classification (i.e., Acc or AUC) (Table 2) .

Next, we will show the performance of the feature selection methods based on four classification methods and two popular evaluation criteria (Acc and AUC) which are widely used in this filed [30,33]. Previous studies have investigated that the number of important genes probably about 50 [17]. To guarantee there are no important genes missing, in the current study, we consider the top 100 features except FCBF because the number of features selected by FCBF is unfixed. With respect to a classifier and a criterion, we also rank the performance for the feature selection methods according to the value they achieved (Table 2). The method with the highest performance of classification will have rank 1, while the worst performance will have rank 7. As shown in Table 2, we notice that LM method obtained better performance than the other feature selection methods for most data sets.

To obtain statistical validation of the results in our experiments, we also use Friedman test [13] to summary the results for feature selection methods over multiple data sets. Friedman is a non-parametric test, which is a promising way to evaluate the performance of methods over multiple data sets [10,11,14,15]. The test statistics *z* for comparing the *i*-th feature selection method and *j*-th feature selection method over multiple data sets can be calculated by the following equation [10,15]

$$z = \frac{\left(R_i - R_j\right)}{SE}.$$

where R_i (or R_j) is the average rank for *i*-th (or *j*-th) feature selection method. *SE* is the standard error in the pairwise comparison between two feature selection methods. In our case, it can be calculated by $SE = \sqrt{\frac{nf(nf+1)}{6{*}nd}} = \sqrt{\frac{7{*}8}{6{*}7}} = 1.115$ (here *nf* and *nd* are the numbers of feature selection method and data sets, respectively). The *z* value is used to calculate the corresponding *p*-value from the table of normal distribution $\mathcal{N}(0, 1)$. Then we can make decision with appropriate level of significance α.

For each criterion (Acc or AUC), we compare the LM with other feature selection methods based on four classifiers (Naive Bayes, KNN, CART, and Random Forest). As shown in Table 3, for FCBF and Relief, we can safely reject their null hypothesis H_0 (there is no difference between two feature selection methods) with significance level $\alpha = 0.05$. For SVMrfe and QPFS, we can not reject the H_0 with $\alpha = 0.05$ when we use Naive Bayes classifier but we can reject the H_0 when we use other three classifiers (KNN, CART, and Random Forest). The competitive methods are mRMR and JMI. Although we can not reject the null hypothesis H_0 of mRMR and JMI with significance level $\alpha = 0.05,$ the probabilities are quite small (the largest *p*-value is 0.155). These results are also consistent with the previous work [6]. Therefore, we can conclude that our method is a reliable method for feature selection problem.

5. Discussion and conclusion

In this article, an optimization model-based global feature selection method is proposed. Compared with QPFS method, our method does not need to consider the positive definiteness of correlation kernel matrix. We only need a symmetrical matrix in model, and it is easy to be satisfied by the transformation $\left(\boldsymbol{K}^T + \boldsymbol{K}\right)/2$ if $\boldsymbol{K}$ is asymmetrical. Another trick in our work is using Lagrange multipliers to select features according to their magnitudes inspired from its practical applications. To our knowledge, it is the first time that the Lagrange multiplier is used to select the discriminative features. The results demonstrate that our method is a reliable feature selection method.

There are also several challenges in feature selection problem [5]. In future work, we will focus on two directions for this problem. On the one hand, convex optimization for feature selection problem has been attracted considerable attention in recent years [4,27]. An interesting line of future work is to develop an efficient and effective algorithm for mathematical model, especially for microarray data analysis they are always involving the large-scale matrix in the model. On the other hand, it is well-known that the drawback of filter methods is how to determine the threshold value in selection process. In our case we only select the top 100 features according to their score. In general, the sparse model are usually adding some constraints (prior distribution) in model to result in sparse coefficients, such as least absolute shrinkage and selection operator(LASSO) [44,49]. How to add some constraints to obtain the sparse solution of dual problem, i.e. sparse Lagrange multipliers, is an interesting direction.

Acknowledgments

We would like to acknowledge support for this project from the National Natural Science Foundation of China (Grant numbers 61173111, 60774086) and the Ph. D. Programs Foundation of Ministry of Education of China (Grant number 20090201110027). We are thankful to the comments from the group in University of Michigan and that greatly improved the manuscript.

References

[1] U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, A.J. Levine, Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, Proceedings of the National Academy of Sciences of the United States of America 96 (12) (1999) 6745–6750.

[2] P. Bermejo, L. de la Ossa, J.A. Gamez, J.M. Puerta, Fast wrapper feature subset selection in high-dimensional datasets by means of filter re-ranking, Knowl. Based Syst. 25 (1) (2012) 35–44.

[3] D.P. Bertsekas, Nonlinear Programming, 2nd, Athena Scientific, USA, 1999.

[4] M. Bogdan, E. van den Berg, C. Sabatti, W. Su, E.J. Candès, Slopeadaptive variable selection via convex optimization, Ann. Appl. Statis. 9 (3) (2015) 1103.

[5] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, Recent advances and emerging challenges of feature selection in the context of big data, Knowl. Based Syst. 86 (2015) 33–45.

[6] G. Brown, A. Pocock, M.J. Zhao, M. Luján, Conditional likelihood maximisation: a unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (1) (2012) 27–66.

[7] Z. Chen, C. Wu, Y. Zhang, Z. Huang, B. Ran, M. Zhong, N. Lyu, Feature selection with redundancy-complementariness dispersion, Knowl. Based Syst. 89 (2015) 203–217.

[8] J.H. Cho, D. Lee, J.H. Park, I.B. Lee, New gene selection method for classification of cancer subtypes considering within-class variation, FEBS Lett. 551 (1–3) (2003) 3–7.

[9] K.A. Clements, A.S. Costa, Topology error identification using normalized lagrange multipliers, IEEE Trans. Power Syst. 13 (2) (1998) 347–353.

[10] J. Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (2006) 1–30.

[11] J. Derrac, S. García, S. Hui, P.N. Suganthan, F. Herrera, Analyzing convergence performance of evolutionary algorithms: a statistical approach, Inf. Sci. 289 (2014) 41–58.

[12] S.D. Flám, H.T. Jongen, O. Stein, Slopes of shadow prices and lagrange multipliers, Optim. Lett. 2 (2) (2008) 143–155.

[13] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, J Am. Atatistical Assoc. 32 (200) (1937) 675–701.

[14] S. García, A. Fernández, J. Luengo, F. Herrera, Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: experimental analysis of power, Inf. Sci. 180 (10) (2010) 2044–2064.

[15] S. García, F. Herrera, An extension on “statistical comparisons of classifiers over multiple data sets” for all pairwise comparisons, J. Mach. Learn. Res. (2008) 2677–2694.

[16] M. Giaquinta, S. Hildebrandt, Calculus of Variations I: The Lagrangian Formalism, 1, Springer, 1996.

[17] T.R. Golub, D.K. Slonim, P. Tamayo et al, Molecular classification of cancer: class discovery and class prediction by gene expression monitoring, Science 286 (5439) (1999) 531–537.

[18] G.J. Gordon, R.V. Jensen, L.L. Hsiao et al, Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma, Cancer Res. 62 (17) (2002) 4963–4967.

[19] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Mach. Learn. 46 (1-3) (2002) 389–422.

[20] M. Ichino, J. Sklansky, Optimum feature-selection by zero-one integer programming, IEEE Trans. Syst. Man Cybern. 14 (5) (1984) 737–746.

[21] G. Isabelle, Andr, Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.

[22] K. Ito, K. Kunisch, Lagrange multiplier approach to variational problems and applications, 15, SIAM, 2008.

[23] K. Kira, L.A. Rendell, The feature selection problem: traditional methods and a new algorithm, in: AAAI, 2, 1992, pp. 129–134.

[24] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artificial Intell. 97 (1-2) (1997) 273–324.

[25] K. Krishnan, J.E. Mitchell, A semidefinite programming based polyhedral cut and price approach for the maxcut problem, Comput. Optim. Appl. 33 (1) (2006) 51–71.

[26] C. Lazar, J. Taminau, S. Meganck, D. Steenhoff, A. Coletta, C. Molter, V. de Schaetzen, R. Duque, H. Bersini, A. Nowe, A survey on filter techniques for feature selection in gene expression microarray analysis, IEEE/ACM Trans. Comput. Biol. Bioinf. 9 (4) (2012) 1106–1119.

[27] H.A. Le Thi, H.M. Le, T.P. Dinh, Feature selection in machine learning: an exact penalty approach using a difference of convex function algorithm, Mach. Learn. 101 (1–3) (2015) 163–186.

[28] Y. Lei, L. Huan, Efficient feature selection via analysis of relevance and redundancy, J. Mach. Learn. Res. 5 (2004) 1205–1224.

[29] H. Liu, H. Motoda, Computational methods of feature selection, CRC Press, 2007.

[30] L. Nanni, S. Brahnam, A. Lumini, Combining multiple approaches for gene microarray classification, Bioinformatics 28 (8) (2012) 1151–1157.

[31] X.V. Nguyen, J. Chan, S. Romano, J. Bailey, Effective global approaches for mutual information based feature selection, in: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, in: KDD ’14, ACM, New York, NY, USA, 2014, pp. 512–521.

[32] H.C. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Trans. Pattern Anal. Mach. Intell. 27 (8) (2005) 1226–1238.

[33] Y. Piao, M. Piao, K. Park, K.H. Ryu, An ensemble correlation-based gene selection algorithm for cancer classification with gene expression data, Bioinformatics 28 (24) (2012) 3306–3315.

[34] S.L. Pomeroy, P. Tamayo, M. Gaasenbeek et al, Prediction of central nervous system embryonal tumour outcome based on gene expression, Nature 415 (6870) (2002) 436–442.

[35] A. Rakotomamonjy, Variable selection using svm based criteria, J. Mach. Learn. Res. 3 (2003) 1357–1370.

[36] I. Rodriguez-Lujan, R. Huerta, C. Elkan, C.S. Cruz, Quadratic programming feature selection, J. Mach. Learn. Res. 11 (2010) 1491–1516.

[37] Y. Saeys, I. Inza, P. Larranaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (19) (2007) 2507–2517.

[38] L.E.A.D. Santana, A.M.D. Canuto, Filter-based optimization techniques for selection of feature subsets in ensemble systems, Expert Syst. Appl. 41 (4) (2014) 1622–1631.

[39] S. Senthamarai Kannan, N. Ramaraj, A novel hybrid feature selection via symmetrical uncertainty ranking based local memetic search algorithm, Knowl. Based Syst. 23 (6) (2010) 580–585.

[40] M.A. Shipp, K.N. Ross, P. Tamayo et al, Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning, Nat. Med. 8 (1) (2002) 68–74.

[41] D. Singh, P.G. Febbo, K. Ross et al, Gene expression correlates of clinical prostate cancer behavior, Cancer Cell 1 (2) (2002) 203–209.

[42] Q. Song, J. Ni, G. Wang, A fast clustering-based feature subset selection algorithm for high-dimensional data, Knowl. Data Eng. IEEE Trans. 25 (1) (2013) 1–14.

[43] S.Q. Sun, Q.K. Peng, A. Shakoor, A kernel-based multivariate feature selection method for microarray data classification, PLoS One 9 (7) (2014) e102541.

[44] R. Tibshirani, Regression shrinkage and selection via the lasso, J. Royal Statis. Soc. Series B (Methodological) (1996) 267–288.

[45] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[46] J. Vergara, P. Estvez, A review of feature selection methods based on mutual information, Neural Comput. Appl. 24 (2014) 175–186.

[47] Y.X. Wang, J.G.M. Klijn, Y. Zhang et al, Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer, The Lancet 365 (9460) (2005) 671–679.

[48] H.H. Yang, J.E. Moody, Data visualization and feature selection: new algorithms for nongaussian data., in: NIPS, 99, Citeseer, 1999, pp. 687–693.

[49] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, J. Royal Statis. Soc. 67 (2) (2005) 301–320.