A Kernel-Based Multivariate Feature Selection Method for Microarray Data Classification

Shiquan Sun, Qinke Peng*, Adnan Shakoor

Systems Engineering Institute, School of Electronic and Information Engineering, Xi’an Jiaotong University, Xi’an, China

Abstract

High dimensionality and small sample sizes, and their inherent risk of overfitting, pose great challenges for constructing efficient classifiers in microarray data classification. Therefore a feature selection technique should be conducted prior to data classification to enhance prediction performance. In general, filter methods can be considered as principal or auxiliary selection mechanism because of their simplicity, scalability, and low computational complexity. However, a series of trivial examples show that filter methods result in less accurate performance because they ignore the dependencies of features. Although few publications have devoted their attention to reveal the relationship of features by multivariate-based methods, these methods describe relationships among features only by linear methods. While simple linear combination relationship restrict the improvement in performance. In this paper, we used kernel method to discover inherent nonlinear correlations among features as well as between feature and target. Moreover, the number of orthogonal components was determined by kernel Fishers linear discriminant analysis (FLDA) in a self-adaptive manner rather than by manual parameter settings. In order to reveal the effectiveness of our method we performed several experiments and compared the results between our method and other competitive multivariate-based features selectors. In our comparison, we used two classifiers (support vector machine, k-nearest neighbor) on two group datasets, namely two-class and multi-class datasets. Experimental results demonstrate that the performance of our method is better than others, especially on three hard-classify datasets, namely Wang’s Breast Cancer, Gordon’s Lung Adenocarcinoma and Pomeroy’s Medulloblastoma.

Introduction

Microarray gene expression based cancer classification is one of the most important tasks in bioinformatics. A typical classification task is to separate healthy patients from cancer patients, based on their gene expression “profile”. However, because cancers are usually marked by changing in the expression levels of certain genes [1], therefore it is obvious that not all measured features are discriminative features for target. Hence, feature selection problem is ubiquitous in cancer classification.

Feature selection techniques for microarray data can be broadly grouped into three categories that are wrapper (classifier-dependent) methods [2,3], embedded (classifier-dependent) methods [4,5] and filter (classifier-independent) methods [6,7]. The primary distinguishing factors among them are computational complexity and the chance of overfitting [8]. Generally, in terms of computational cost, filters are faster than embedded methods, which are in turn faster than wrappers. In terms of overfitting, wrappers have higher learning capacity so are more likely to overfit than embedded methods, which in turn are more likely to overfit than filter methods [9]. Filter methods can be divided into two classes, univariate-based filters and multivariate-based filters [8]. Univariate filter methods have attracted much attention because of their low complexity and fast performance for high dimensionality of microarray data analyses. However, some valuable genes discarded by univariate methods may have great contribution for classification [10]. Therefore, the major reason of their less accurate performance is that they disregard the effects of feature-feature (we use without distinction the term “feature” and “gene” in the paper) interactions. The applications of multivariate filter methods are simple bivariate-based methods which are almost based on entropy (or conditional entropy) and mutual information [9,11], such as mRMR [7,12], CFS [13] and several variants of the Markov blanket filter method [14]. However, they also abandon presumably redundant variables that can result in a performance loss [15].

Partial least squares (denoted as PLS), which shares the characteristics of other regression and feature transformation techniques (such as canonical correlation analysis and principal component analysis), has proven to be useful in situations when the number of observed variables (D) are significantly greater than the number of observations (N) (e.g., N ≪ D). In other words, PLS is a popular approach to solve problems when there is high multicollinearity among features [16]. SlimPLS [17], PLSRFE [18,19] and TotalPLS [20] are multivariate-based feature selection methods that were proposed by Gutkin et al. and You et al.,
In order to state dot product operation in the algorithm, we can restrict \( v \) to belong to the linear spans of the points. They can therefore be expressed as:

\[
\vec{v} = (\Phi(x_1), \cdots, \Phi(x_N))^{T} \beta^\Phi
\]

\[
\begin{bmatrix}
\Phi(x_1) \\
\vdots \\
\Phi(x_N)
\end{bmatrix}
\begin{bmatrix}
\Phi(x_1) \\
\vdots \\
\Phi(x_N)
\end{bmatrix}^{T} = K_{X} \beta^\Phi
\]

Let \( K_{X}(x_i, x_j) \) be an element of the Gram matrix \( K_{X} \) in feature space and \( h \) is the desired number of components. Deflating \( Y \) will, however, be needed for kernel partial least squares.

The first component for kernel PLS can be determined as eigenvector of the following square kernel matrix for \( \beta^\Phi \):

\[
\beta^\Phi x_1 = K_{X} \beta^\Phi
\]

where \( \lambda \) is an eigenvalue. The size of the kernel matrix \( K_{X} \) is \( N \times N \). Hence, no matter how many variables there are in the original matrices \( X \) and \( Y \), the size of these kernel matrices will not be affected by it. Therefore, the combination of PLS with kernel produces a powerful algorithm that will solve this problem rapidly and effectively. The geometric representation of kernel PLS can be found in Figure 1(b). The kernel PLS algorithm procedure and the number of determined components can be found in Table 1 (https://github.com/sqsun/kernelPLS).

**The importance of each feature**

In original space, let \( T \) be a set of components, \( T = \{ t_1, t_2, \cdots, t_h \} \). The accumulation of variation explanation of \( T \) to \( Y \) is given by [24,25]

\[
w_i = \sqrt{ \sum_{j=1}^{h} \frac{\Psi(Y_{j}, t_i)^2}{\sum_{j=1}^{h} \Psi(Y_{j}, t_i)}}
\]

where \( h \) is the number of components and \( w_i \) is the weight of the \( i \)th feature for the \( h \)th component. \( \Psi(Y_{j}, t_i) = \sum_{j=1}^{C} \Psi(y_{j}, t_i) \) is the correlation between \( t_i \) and \( Y_j \), where \( \Psi(\cdot, \cdot) \) is correlation function. The larger value of \( w_i \), the more explanatory power of the \( i \)th feature to \( Y \).

It is worth noting that the above equation can also be used in kernel space. The reason is holding of equation \( \Phi(y_j) = y_j \), because here \( y_j \) is class label. So the expression \( \Psi(\Phi(y_j), t_i^\Phi) \) can be expressed as \( \Psi(y_j, t_i^\Phi) \), here \( t_i^\Phi \in T^\Phi \) and \( T^\Phi = \{ t_1^\Phi, t_2^\Phi, \cdots, t_h^\Phi \} \).

**Model selection**

Two issues are still unresolved before applying kernel PLS for feature selection. The number of components and the number of features are unknown.

**The number of components**

In order to determine the number of components \( h \), there are two widely used methods in the previous works, one is setting a fixed number, such as \( h = 3 \), and another is by cross validation (CV). Different datasets contain various data structures, therefore, a fixed number is not suitable for all datasets. Although the CV
combined with various classifiers lead to good performance, it suffers from huge computational burden.

To fully circumvent these difficulties, [26] has given an implication of close relationship between PLS and Fisher’s linear discriminant analysis (FLDA) in original space. FLDA can be considered as an optimization problem $\arg\max_{x \in \mathbb{R}^d} \{x^T S_1 x / x^T S_2 x\}$, e.g. finding an appropriate projection vector $x$. Where $S_1$ presents the inter-class scatter matrix, $S_2$ are the intra-class scatter matrix and the intra-class scatter matrix in kernel space, respectively. We consider

$$\gamma_l = \frac{\sum_{i=1}^{C} N_i m_{il}^0}{\sum_{i=1}^{C} N_i}$$

It denotes the contribution of the $l$th component for classification. Where $N_i$ indicates the number of samples in the $i$th class, here $m_{il}^0$ represents the mean vector of the $l$th class with respect to $l$ th component in projection space and the $\gamma_l$ represents segmentation threshold of classification, the larger $\gamma_l$ corresponds to the more significant in classification.

The number of features

Figure 2 shows how classification performance varies with the change in number of features which were selected. The average classification error rate was calculated by two classifiers on all test datasets. An improvement in performance could be evident if the number of related features increase from 1 to 25, but after increasing number of features beyond 25, no significant improvement was obvious. In order to find optimum results for all the datasets, we extend the range from 20 to 50 features configurations in our study.

### Results

#### Test datasets

To assess the performance of our method, we have conducted several experiments on a number of publicly available datasets. Summary of all data sets we used in our experiments can be found in table 2 and following is the brief description of each data set.

- **AMLALL(A)**([27]). There are two parts containing the initial (train), 38 bone marrow samples from two classes: 27 cases of acute lymphoblastic leukemia(ALL) and 11 cases of acute myeloid leukemia(AML); independent (test), 34 samples from two classes: 20 cases of ALL and 14 cases of AML. Each case is described by expression levels of 7129 probes from 6817

---

**Table 1. Algorithm 1: kernelPLS.**

| Input | $K_k$ - kernel matrix |
|-------|-----------------------|
| $K_l$ - kernel matrix |
| Output | $w$ - the weight of each feature |

1: Initializing $K_l \leftarrow K_k, \gamma_1 = + \infty$;
2: $\varepsilon = 0.01, I = 1$;
3: while $\gamma_I > \varepsilon$ do;
4: $\beta_0^I, \beta_1^I$;
5: while $|\beta_0^I - \beta_1^I| > \varepsilon$ do;
6: $\beta_0^I - K_l \beta_1^I$;
7: $\beta_0^I - \frac{\beta_1^I}{\varepsilon}$;
8: end while;
9: Calculating the component $\beta_0^I, \beta_1^I - K_l \beta_1^I$;
10: Deflating target matrix $Y_0, Y_{I+1} = Y_0 - \Delta^{-1} \beta_0^I \beta_1^T Y_0$, where $\Delta = \beta_0^I \beta_1^T$;
11: Deflating kernel matrix $K_0, K_{I+1} = (I - \Delta^{-1} \beta_0^I \beta_1^T) K_0 (I - \Delta^{-1} \beta_0^I \beta_1^T)$;
12: Calculating the contribution of the $l$th component $\gamma_l$ = $\sum_{i=1}^{M} N_i m_{il}^0$ $\sum_{i=1}^{M} N_i$;
13: $I = I + 1$;
14: end while;
15: $h = I - 1$;
16: Calculating the weight of each feature $w$ via Equation(1);
17: return $w$
human genes. Source: http://www-genome.wi.mit.edu/cgi-bin/cancer/datasets.cgi;

**Breast**(B) ([28]). The dataset used the raw intensity Affymetrix CEL files and normalized the data by RMA procedures. A final expression matrix comprising 22283 features and 209 samples, 71 of which are from patients, the rest 138 samples are normal samples. Source: http://math.bu.edu/people/sray/software/prediction;

**Lung**(L) ([29]). This dataset contains 86 samples: 24 are tumor samples and 62 are normal controls, 7129 genes with highest intensity across the samples are considered. Source: http://math.bu.edu/people/sray/software/prediction/;

**Prostate**(P) ([30]). This dataset contains 52 prostate tumor samples and 50 normal samples with 12600 genes. An independent set of testing samples is generated from the training data, 25 tumor and 9 normal samples are extracted

![Figure 2. The effect of different numbers of selected features](image)

**Figure 2. The effect of different numbers of selected features.** Two classifier, SVM and KNN, are used for measuring the performance of average error of all test datasets based on kernelPLS selector. Where the optimal parameters of RBF kernel SVM are determined by partial swarm optimization and the parameter $k$ for the nearest neighbors is 5.

doi:10.1371/journal.pone.0102541.g002

### Table 2. The cancer classification datasets¹ used in the present paper.

| Class   | Dataset | Sample | Feature | Class | Source |
|---------|---------|--------|---------|-------|--------|
| Two-class | AMLALL | 72     | 7129    | 2     | [27]   |
|         | Breast  | 209    | 22283   | 2     | [42]   |
|         | Lung    | 86     | 7129    | 2     | [29]   |
|         | Prostate| 102    | 12600   | 2     | [30]   |
|         | DLBCL   | 77     | 7129    | 2     | [31]   |
|         | Medulloblastoma | 60 | 7129 | 2 | [32] |
| Multi-class | SJUDE | 215     | 12358   | 7     | [13]   |
|         | Lymphoma| 62     | 4026    | 3     | [33]   |
|         | SRBCT   | 83     | 2308    | 4     | [34]   |
|         | MLL     | 72     | 8685    | 3     | [35]   |
|         | Lung    | 203    | 3312    | 5     | [37]   |

¹Available at https://github.com/sqsun/kernelPLS-datasets.
doi:10.1371/journal.pone.0102541.t002
Table 3. Description of genes reported by existing published papers and ranked by our method.

| Accession number | Gene description     | References | Rank |
|------------------|----------------------|------------|------|
| X95735_at        | Zyxin                | [43] [38] [27] [44] [28] | 4    |
| M23197_at        | CD33                 | [43] [38] [27] [44] [28] | 8    |
| U22376_cds2_s_at | C-myb                | [38] [27] [44] [28]     | 74   |
| M27891_at        | Cystatin C           | [43] [38] [27] [44] [28] | 21   |
| M16038_at        | LYN                  | [38] [27] [44] [28]     | 11   |
| M84526_at        | DF(adipsin)          | [43] [38] [27] [44]     | 9    |
| M27783_s_at      | ELA2 Elastatse 2     | [38] [44] [28]          | 80   |
| U50136_mal1_at   | LTC4 synthsne        | [38] [27] [28]          | 3    |
| Y12670_at        | Leptin receptor      | [38] [27] [28]          | 2    |
| U46499_at        | Glutathione          | [43] [38] [44]          | 96   |
| L09209_s_at      | Amyloid beta         | [43] [38] [44]          | 48   |
| U46751_at        | p62                  | [38] [27]               | 19   |
| M55150_at        | Fumarylacetoacetate  | [38] [27]               | 7    |
| M38652_s_at      | Properdin            | [38] [27]               | 22   |
| M80254_at        | CyP3                 | [27] [28]               | 17   |
| X17042_at        | Proteoglycan 1       | [43] [27]               | 10   |
| U82759_at        | HoxA9                | [43] [27]               | 8    |

doi:10.1371/journal.pone.0102541.t003

A Kernel-Based Multivariate Feature Selection Method

Comparison of selected genes

In our first experiment, we used two datasets, namely the Leukemia data (two-class) of [27] and the Lymphoma data (three-class) of [33], to compare our method with previous works with respect to the selected genes.

For the Leukemia data, we collected several most important genes in Table 3 that were published in several papers. It can readily be seen that three probes, X95735_at, M27891_at and M23197_at were reported by five published papers, and their ranking by our method are 4th, 17st and 8st, respectively. We notice that there are many overlapping of genes among the list of papers.

For Leukemia data, the top-ranked 40 features obtained by our procedure are shown in Table 4 in which genes are in columns from 1 to 40. There is a worthwhile result achieved by our
method, that is, it obtained the genes with the highest weight. Many of these genes are known as differentially expressed genes by many foregoing studies. 24 out of 40 genes are listed in this table that were also selected by [27], which shows the effectiveness of our method.

For the Lymphoma data of [33], the missing values are imputed by KNN-imputed method (k = 10). The top 40 genes ranked by our procedure are listed in Table 5. From the table, we can see that important genes can be captured easily by our method. There are many genes that are also chosen by [38].

Figure 3 illustrates the differentially expressed genes for two datasets, namely the Leukemia data and the Lymphoma data. No single gene is uniformly expressed across the class, all these genes as a group appear correlated with class which is illustrating the effectiveness of the Kernel PLS method. In Figure 3(a) the top panel is consist of three genes GENE1622X, GENE2402X and GENE1648X that are highly expressed in DLCL, middle panel is comprised of GENE1606X, GENE896X and GENE1617X that are highly expressed in DLCL but moderately expressed in FL. Bottom panel compose of three genes, namely GENE1606X, GENE681X and GENE1618X, are highly expressed in CLL. In Figure 3(b) the top panel shows three probes highly express in AML and the bottom panel shows three probes more highly expression in ALL. The probe U377055_rna1_s_at was found by our method to distinguish AML from ALL. Figure 3(c) demonstrate the projected result of top 100 genes using sammon mapping which shows DLBCL, CLL, FL are very clear and the boundaries can be easily drawn.

Comparison of several multivariate-based feature selectors

In our second experiment, we compared several feature selectors with our procedure based on two classifiers, SVM and KNN. In our experiments, we choose the RBF kernel for each dataset to perform classification. To determine the best values of $C$ and $\gamma$, we conducted particle swarm optimization algorithm to pick the pair $(C, \gamma)$ with best accuracy in the range of $C \in \{10^{-3}, \ldots, 10^{2}\}$ and $\gamma \in \{10^{-3}, \ldots, 10^{4}\}$. We set the parameter to $k=5$ for $k$-nearest neighbor. To obtain a statistically reliable predictive measure, we performed 10-fold cross validation for two-class datasets and 5-fold cross validation for multi-class datasets. The results are evaluated by classification accuracy (Acc), area under receiver operating characteristic curve (AUC) for two-class problems and classification accuracy (Acc), Cohen’s Kappa coefficient (kappa) for multi-class problems. The reason of using 5-fold cross validation for multi-class datasets is that there is just a few number of samples in some groups (classes) of these datasets. Therefore to ensure the presence of samples of each class in training and also in test datasets we need to perform 5-fold cross validation for multi-class datasets.

In this paper, the comparison was conducted with four competitive algorithms, PLS, ReliefF, SVMrfe and mRMR. The

### Table 4. Top-ranked 40 features selected using kernelPLS for the Leukemia dataset.

| Rank | Gene Symbol   | Rank | Gene Symbol   | Rank | Gene Symbol   |
|------|---------------|------|---------------|------|---------------|
| 1.   | M23197_at     | 11.  | M16038_at     | 21.  | M27891_at     |
| 2.   | Y12670_at     | 12.  | M96326_rna1_at| 22.  | M83652_s_at   |
| 3.   | US0136_rna1_at| 13.  | X70297_at     | 23.  | M19507_at     |
| 4.   | X95735_at     | 14.  | M62762_at     | 24.  | M63138_at     |
| 5.   | D49950_at     | 15.  | X58116_rna1_s_at| 25.  | X58431_rna2_s_at| 26.  | Y00787_s_at       |
| 6.   | X50485_rna1_at| 16.  | L08246_at     | 27.  | M83652_at     |
| 7.   | M55150_at     | 17.  | M80254_at     | 28.  | X52056_at     |
| 8.   | U82759_at     | 18.  | M22960_at     | 29.  | M11147_at     |
| 9.   | M84526_at     | 19.  | U46751_at     | 30.  | M11147_at     |
| 10.  | X17042_at     | 20.  | M81933_at     | 31.  | M57710_at     |

1. The boldfaced genes were selected by [38].

doi:10.1371/journal.pone.0102541.t004

### Table 5. Top-ranked 40 features selected using kernelPLS for the Lymphoma dataset.

| Rank | Gene Symbol   | Rank | Gene Symbol   | Rank | Gene Symbol   |
|------|---------------|------|---------------|------|---------------|
| 1.   | GENE1622X     | 11.  | GENE1608X     | 21.  | GENE1636X     |
| 2.   | GENE2403X     | 12.  | GENE622X      | 22.  | GENE710X      |
| 3.   | GENE653X      | 13.  | GENE833X      | 23.  | GENE2401X     |
| 4.   | GENE1644X     | 14.  | GENE712X      | 24.  | GENE1641X     |
| 5.   | GENE1607X     | 15.  | GENE735X      | 25.  | GENE654X      |
| 6.   | GENE1647X     | 16.  | GENE1553X     | 26.  | GENE1661X     |
| 7.   | GENE1610X     | 17.  | GENE078X      | 27.  | GENE1702X     |
| 8.   | GENE2402X     | 18.  | GENE530X      | 28.  | GENE642X      |
| 9.   | GENE1648X     | 19.  | GENE721X      | 29.  | GENE1744X     |
| 10.  | GENE1643X     | 20.  | GENE2400X     | 30.  | GENE689X      |

1. The boldfaced genes were selected by [38].

doi:10.1371/journal.pone.0102541.t005
parameter setting of them are as follows: for the PLS-based feature selection, we used the SIMPLS method and the number of components determined by self-adaptive manner which is the same as the kernelPLS (the proposed method). The parameter $k$ of ReliefF is equal to the number of sample according to the published paper [39]. For SVMrfe, in order to ensure acceptable running time, we use SVM with RBF kernel and its parameter settings are same as in LIBSVM.

Without loss of generality, we used two datasets, Breast(two-class) and Lymphoma(three-class) to show the performance of our method. Figure 4 shows the comparison of error rate between our method and four other methods. One can see that when number of selected features are 30, error rate of our method is less than other methods for both classifiers and both datasets.

Table 6 and 7 summarized the comparison of results generated by our method and other methods with respect to Acc and AUC for two-class datasets. From the results, we can see that the performance of our method is better than others. Refers to table 6 we can see that for Breast(B) and Prostate(P) datasets, accuracy of our method is considerably higher as compare to other methods, which shows the effectiveness of our method.

Similarly in table 7 for datasets Breast, Lung, DLBCL, Medulloblastoma, Prostate and Stjude, kernelPLS shown better accuracy rate for SVM classifier weather than KNN. Both Acc and AUC values of our method have higher values among others and finally the average results likewise are best. Although for few datasets our results are similar to their results but in these cases time taken by our method is significantly smaller than other methods. For example in table 7 for AMLALL dataset, including our method, the AUC is 100% for many methods while time consumed by our method is only 0.0891 s while the time taken by other methods, ReliefF, mRMR, SVMrfe and PLS, are about 5 s, 52 s, 210 s and 12 s, respectively. So time consumption by our algorithm is many times less than others which depicts overall well performance of our method.

It is worth noting that our method outperforms others on three hard-classify datasets, Wang's Breast cancer, Gordon’s Lung adenocarcinoma and Pomeroy’s Medulloblastoma. We also make a comparison with the results of other feature selectors in published papers. For example, the reference [40] reported that the accuracies of $k$-TSP+SVM on these datasets were 67.1%, 72.2% and 64.2%, respectively. The reference [41] combined multiple feature selection (or feature transform) approaches for Medulloblastoma dataset and the obtained highest Acc was 70%.

To estimate the performance of our method we did not limit our evaluation to only two-class datasets we also used 5 multi-class datasets in our experiments. Tables 8 and 9 demonstrate the comparison of kernelPLS with other methods for multi-class datasets on the bases of results obtained for two evaluation measures, namely Acc and Kappa. Results shown in table 8 and table 9 are for two classifiers KNN and SVM, respectively. In table 8 results obtained by kernelPLS are better than Relief, SVMrfe and PLS and highly competitive to mRMR method for several multi-class datasets. For example in case of Stjude dataset for Acc and Kappa values by kernelPLS are 96.4% and 0.956 respectively which are highest among all values achieved by other methods. Likewise table 9 authenticates the high performance by kernelPLS over other methods for SVM classifier. Here one can see that kernelPLS give outperforming results for all datasets by achieving accuracies and Kappa coefficients values superior than all other methods. As a conclusion the overall high average Acc and Kappa values in both tables show the effectiveness and significance of our method as compare to other popular methods.

Table 10 shows the comparison between running time taken by our method and other methods. There is no single method among these that can perform faster than our method. It clearly shows that kernelPLS is faster than the other algorithms. For example for
AMLALL dataset time consumed by our method is 0.0891 s while time spent by ReliefF, mRMR, SVMrfe and PLS are 5.1510 s, 52.5854 s, 210.4046 s, 12.1222 s, respectively.

Discussion

In this article, we proposed an effective multivariate-based feature filter method for cancer classification, namely, kernelPLS-based filter method. We showed that gene-gene interactions cannot be ignored in feature selection techniques to improve classification performance. In other words the nonlinear relationship of gene-gene interactions is a vital concept that can be taken into account to enhance accuracy. To capture these nonlinear relations of interaction between genes we used kernel method because kernel method can be used to reveal the intrinsic relationships that are hidden in the raw data. In order to capture the reasonable number of components, we make use of the relationship between PLS and linear discriminant analysis to determine the number of components in kernel space based on kernel linear discriminant analysis. To verify the importance of gene-gene interactions we compared our feature selector with other multivariate-based feature selection methods by using two classifiers SVM and KNN. Experimental results, expressed as both accuracy(Acc) and area under the ROC curve(AUC), showed that our method leads to promising improvement in ACC and AUC.

We can conclude that the gene-gene interactions whats more, nonlinear relationships of gene-gene interactions are core interactions that can improve classification accuracy, efficiently. We can summarize the characteristics of proposed approach as follows: (1)Fast and efficient. The time complexity of deflation procedure used after the extraction of each component scale is $O(N^2)$, where $N$ is the number of sample. In most cases, the number of sample in microarray data is less than 150, therefore, the running speed of kernelPLS procedure(feature selection time) is faster than others, which are summarized in table 10. (2)Model-free, e.g. no need the distributional assumptions. Because of small sample size, it is difficult to validate distributional assumptions, such as Gaussian distribution, Gamma distribution etc. (3)Applicable to both two-class as well as multi-class classification problems.

In our method, the choice of kernel functions can affect the results. When high dimensionality exist(such as microarray datasets), the performance of linear kernel is better than Gauss
### Table 6. Comparison of kernelPLS with four other feature filters for the classification accuracy(%) and AUC(%) of KNN(k = 5) on two-class datasets.

| Dataset | ReliefF Acc | ReliefF AUC | mRMR Acc | mRMR AUC | SVMrfe Acc | SVMrfe AUC | PLS Acc | PLS AUC | kernelPLS Acc | kernelPLS AUC |
|---------|-------------|-------------|----------|----------|-----------|-----------|--------|--------|-------------|-------------|
| A       | 96.1        | 98.3        | 97.5     | 98.3     | 98.8      | 99.0      | 90.7   | 97.3   | 94.6        | 99.0        |
| B       | 68.5        | 66.5        | 67.9     | 67.7     | 68.5      | 67.2      | 69.9   | 70.2   | 71.8        | 75.5        |
| L       | 74.2        | 77.4        | 74.2     | 74.2     | 74.3      | 75.5      | 75.7   | 76.5   | 73.2        | 78.3        |
| D       | 93.8        | 97.5        | 95.0     | 99.6     | 93.4      | 98.4      | 91.1   | 96.3   | 95.0        | 97.1        |
| M       | 70.0        | 73.5        | 71.7     | 77.3     | 65.0      | 68.8      | 73.3   | 80.8   | 73.3        | 76.2        |
| P       | 95.0        | 98.1        | 96.0     | 96.6     | 90.1      | 92.3      | 95.3   | 98.3   | 96.0        | 98.9        |
| Avg.    | 82.9        | 85.2        | 83.7     | 85.6     | 81.7      | 83.5      | 82.7   | 86.6   | 84.0        | 87.5        |

DOI: 10.1371/journal.pone.0102541.t006

### Table 7. Comparison of kernelPLS with four other methods. For 10-fold cross validation classification accuracy(%) and AUC (%) of SVM on two-class datasets.

| Dataset | ReliefF Acc | ReliefF AUC | mRMR Acc | mRMR AUC | SVMrfe Acc | SVMrfe AUC | PLS Acc | PLS AUC | kernelPLS Acc | kernelPLS AUC |
|---------|-------------|-------------|----------|----------|-----------|-----------|--------|--------|-------------|-------------|
| A       | 97.5        | 100         | 96.3     | 100      | 97.5      | 100       | 94.6   | 100   | 96.1        | 100         |
| B       | 68.0        | 69.2        | 69.9     | 67.5     | 69.9      | 67.2      | 72.2   | 71.5   | 72.7        | 75.4        |
| L       | 77.4        | 81.5        | 72.1     | 76.5     | 73.3      | 75.8      | 76.8   | 77.6   | 77.4        | 82.6        |
| D       | 94.8        | 98.2        | 94.8     | 99.2     | 93.4      | 99.4      | 93.4   | 98.3   | 97.5        | 100         |
| M       | 71.7        | 72.9        | 70.0     | 73.1     | 66.7      | 69.7      | 70.0   | 77.2   | 73.3        | 82.7        |
| P       | 96.0        | 97.5        | 96.0     | 96.7     | 89.1      | 94.2      | 95.1   | 98.7   | 97.3        | 97.9        |
| Avg.    | 84.2        | 86.7        | 83.2     | 85.5     | 81.7      | 84.8      | 83.7   | 87.2   | 85.7        | 89.8        |

DOI: 10.1371/journal.pone.0102541.t007
Table 8. Comparison of kernelPLS with four other feature filters for the classification accuracy(%) and Cohen’s kappa coefficient of KNN(k = 5) on multi-class datasets.

| Dataset | ReliefF | mRMR | SVMrfe | PLS | kernelPLS |
|---------|---------|------|--------|-----|-----------|
|         | Acc     | Kappa| Acc    | Kappa| Acc       | Kappa | Acc   | Kappa | Acc   | Kappa |
| St      | 83.9    | 0.811| 88.7   | 0.852| 81.9      | 0.797 | 86.9  | 0.842 | 89.9  | 0.876 |
| Ly      | 98.5    | 0.964| 100    | 1    | 98.3      | 0.969 | 100   | 1    | 100   | 1     |
| Lu      | 72.2    | 0.271| 73.3   | 0.403| 73.3      | 0.268 | 76.8  | 0.404 | 76.8  | 0.428 |
| ML      | 87.7    | 0.762| 94.6   | 0.903| 91.7      | 0.852 | 89.0  | 0.794 | 93.1  | 0.877 |
| SR      | 91.6    | 0.884| 98.8   | 0.983| 91.5      | 0.880 | 91.5  | 0.877 | 96.4  | 0.947 |
| Avg.    | 86.8    | 0.738| 91.1   | 0.828| 87.3      | 0.753 | 88.8  | 0.783 | 91.2  | 0.826 |

Table 9. Comparison of kernelPLS with four other methods. For 5-fold cross validation classification accuracy(%) and Cohen’s kappa coefficient of SVM on multi-class datasets.

| Dataset | ReliefF | mRMR | SVMrfe | PLS | kernelPLS |
|---------|---------|------|--------|-----|-----------|
|         | Acc     | Kappa| Acc    | Kappa| Acc       | Kappa | Acc   | Kappa | Acc   | Kappa |
| St      | 86.2    | 0.849| 88.9   | 0.866| 86.4      | 0.851 | 86.8  | 0.834 | 89.9  | 0.876 |
| Ly      | 100     | 1    | 100    | 1    | 96.7      | 0.933 | 100   | 1    | 100   | 1     |
| Lu      | 76.9    | 0.451| 76.9   | 0.399| 74.6      | 0.382 | 74.5  | 0.360 | 79.2  | 0.532 |
| ML      | 94.6    | 0.906| 93.2   | 0.884| 87.7      | 0.801 | 90.3  | 0.834 | 95.8  | 0.919 |
| SR      | 96.4    | 0.947| 98.8   | 0.983| 97.6      | 0.964 | 98.8  | 0.983 | 97.6  | 0.964 |
| Avg.    | 90.8    | 0.831| 91.6   | 0.826| 88.6      | 0.786 | 90.1  | 0.802 | 92.5  | 0.858 |

doi:10.1371/journal.pone.0102541.t008
doi:10.1371/journal.pone.0102541.t009
kernel for our method. What’s more, in case of linear kernel there is no noticeable effect on the results while adjusting its parameters.

### Table 10. The running time(s) of five feature filtering methods on two groups cancer classification datasets.

| Class  | Dataset | Relief | mRMR$^1$ | SVMrfe | PLS | kernelPLS |
|--------|---------|--------|----------|--------|-----|----------|
| Two-class | A | 5.1510 | 52.5854 | 210.4046 | 12.1222 | 0.0891 |
|         | B | 5.1496 | 88.6176 | >1e+003 | 10.6423 | 0.1092 |
|         | L | 7.5420 | 52.8977 | 693.1857 | 16.8629 | 0.2410 |
|         | D | 5.5614 | 53.1088 | 221.2261 | 12.0526 | 0.0965 |
|         | M | 5.1343 | 51.9969 | 421.8250 | 19.2384 | 0.2676 |
|         | P | 18.1848 | 65.1076 | >1e+003 | 64.2148 | 0.6010 |
| Multi-class | St | 34.0030 | 67.5321 | >1e+003 | >1e+003 | 2.1180 |
|         | Ly | 2.7332 | 5.7846 | 217.2568 | 27.9456 | 0.2361 |
|         | Lu | 10.2526 | 9.7816 | >1e+003 | 17.8940 | 0.5500 |
|         | ML | 6.6426 | 8.7484 | 791.0244 | 98.8890 | 0.2586 |
|         | SR | 1.8230 | 5.8336 | 87.6536 | 8.8784 | 0.1714 |

$^1$Time required for selecting 1000 features.


doi:10.1371/journal.pone.0102541.t010

### Author Contributions

Conceived and designed the experiments: SS QP. Performed the experiments: SS. Analyzed the data: SS AS. Contributed reagents/materials/analysis tools: SS. Contributed to the writing of the manuscript: SS AS.

### References

1. Piao Y, Piao M, Park K, Ryu KH (2012) An ensemble correlation-based gene selection algorithm for cancer classification with gene expression data. Bioinformatics 28: 3306–3315.
2. Kohavi R, John GH (1997) Wrappers for feature subset selection. Artificial Intelligence 97: 273–324.
3. Xiong M, Fang X, Zhao J (2001) Biomarker identification by feature wrappers. Genome Research 11: 1676–1687.
4. Senthumari Kannan S, Ramaraj N (2010) A novel hybrid feature selection via symmetrical uncertainty ranking based local memetic search algorithm. Knowledge-Based Systems 23: 580–583.
5. Díaz-Uriarte R, Andrés ÁAd (2006) Gene selection and classification of microarray data using random forest. BMC Bioinformatics 7.
6. Lei Y, Huang L (2004) Efficient feature selection via analysis of relevance and redundancy. Journal of Machine Learning Research 5: 1205–1224.
7. Beng H, Fuhs I, Ding C (2005) Feature selection based on mutual information criteria of maxdependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 27: 1226–1239.
8. Saey Y, Inza I, Larrañaga P (2007) A review of feature selection techniques in bioinformatics. Bioinformatics 23: 2307–2317.
9. Caviness B, Adam P, Ming-Jie Z, Mikel I (2012) Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. Journal of Machine Learning Research 13: 27–66.
10. Issa B, Andr Elisaef (2003) An introduction to variable and feature selection. The Journal of Machine Learning Research 3: 1157–1162.
11. Balagani KS, Ploha YV (2010) On the feature selection criterion based on an approximation of multidimensional mutual information. IEEE Transactions on Pattern Analysis and Machine Intelligence 32: 1342–1343.
12. Dr Jay N, Papillon-Cavanagh S, Olsen C, El-Hachem N, Bontempi G, et al. (2013) meminer: an r package for parallelized meminer feature selection. Bioinformatics 29: 2365–2368.
13. Yeoh EJ, Ross ME, Shurtleff SA, Williams WK, Patel D, et al. (2002) Molecular classification of cancer: class discovery and class prediction by gene expression profiling. Science 290: 1151–1156.
14. You W, Yang Z, Ji G (2014) Feature selection for high-dimensional multi-category data using plb-based local recursive feature elimination. Expert Systems with Applications 41: 1463–1473.
15. You W, Yang Z, Yuan M, Ji G (2014) Totalpls: Local dimension reduction for multicategory microarray data. IEEE Transactions on Human-Machine Systems 44: 125–130.
16. You W (1966) Estimation of principal components and related models by iterative least squares. Multivariate Analysis. New York: Academic.
17. Shawe-Taylor J, Nello (2004) Kernel methods for pattern analysis. UK: Cambridge University.
18. Rannar S, Lindgren F, Geladi P, Wold S (1994) A pls kernel algorithm for data sets with many variables and fewer objects. Part 1: Theory and algorithm. Journal of Chemometrics 8: 111–125.
19. Wold S, Jöhnsson W, Cocchi M (1993) PLS-Partial least-squares projections to latent structures. 3D QSAR in Drug Design, Theory Methods and Applications. Berlin: Springer-Verlag.
20. You W, Ji G, Yuan M (2011) Pls-based gene selection and identification of tumor-specific genes. IEEE Transactions on Systems Man and Cybernetics Part C-Applications and Reviews 41: 830–841.
21. Li GZ, Zhao BW, Qin JY, You M (2012) Model selection for partial least squares based dimension reduction. Pattern Recognition Letters 33: 524–529.
22. Golub TR, Slonim DK, Tamayo P, Huard C, Gaasenbeek M, et al. (1999) Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. Science 286: 337–338.
23. Wang HQ, Wong HS, Huang DS, Shu J (2007) Extracting gene regulation information for cancer classification. Pattern Recognition 40: 3379–3392.
24. Gordon GJ, Jensen RV, Hiaio LL, Guillans SR, Blumenstock JE, et al. (2002) Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. Cancer Research 62: 4963–4967.
25. Singh D, Febbo PG, Ross K, Jackson DG, Manola J, et al. (2002) Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1: 203–209.
26. Ship MA, Ross KN, Tamayo P, Weng AP, Kutok JL, et al. (2002) Diffuse large b-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. Nature Medicine 8: 68–74.
27. Pomeroy SL, Tamayo P, Gaasenbeek M, Sturla LM, Angelo M, et al. (2002) Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature 415: 436–442.
28. Alizadeh AA, Eisen MB, Davis RE, Ma C, Lossos IS, et al. (2000) Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. Nature 415: 436–442.
35. Armstrong SA, Staunton JE, Silverman LB, Pieters R, de Boer ML, et al. (2002) Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. Nature Genetics 30: 41–47.
36. Yang K, Cai ZP, Li JZ, Lin GH (2006) A stable gene selection in microarray data analysis. BMC Bioinformatics 7.
37. Bhattacharjee A, Richards WG, Staunton J, Li C, Monti S, et al. (2001) Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. Proceedings Of the National Academy Of Sciences Of the United States Of America 98: 13790–13795.
38. Draminski M, Rada-Iglesias A, Earoth S, Wadelius C, Koronacki J, et al. (2008) Monte carlo feature selection for supervised classification. Bioinformatics 24: 110–117.
39. Brown G, Pocock A, Zhao MJ, Lujn M (2012) Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. The Journal of Machine Learning Research 13: 27–66.
40. Shi P, Ray S, Zhu Q, Kon MA (2011) Top scoring pairs for feature selection in machine learning and applications to cancer outcome prediction. Bmc Bioinformatics 12.
41. Nanni L, Brahnam S, Lumini A (2012) Combining multiple approaches for gene microarray classification. Bioinformatics 28: 1151–1157.
42. Wang Y, Klijn JGM, Zhang Y, Sieuwerts AM, Look MP, et al. (2005) Gene-expression profiles to predict distant metastasis of lymph-node-negative primary breast cancer. The Lancet 365: 671–679.
43. Chu W, Ghahramani Z, Falciani F, Wild DL (2005) Biomarker discovery in microarray gene expression data with gaussian processes. Bioinformatics 21: 3385–3393.
44. Lee KE, Sha N, Dougherty ER, Vannucci M, Mallick BK (2003) Gene selection: a bayesian variable selection approach. Bioinformatics 19: 90–97.