An adaptive feature selection method for microarray data analysis

Jie Cheng[1], Joel Greshock[2], Leming Shi[3], Jeffery Painter[1], Xiwu Lin[1], Kwan Lee[1], Shu Zheng[4], Richard Wooster[2], Lajos Pusztai[5], Alan Menius[1]

[1]Quantitative Sciences, GlaxoSmithKline, Collegeville, PA 19426, USA
[2]Cancer Research, GlaxoSmithKline, Collegeville, PA 19426, USA
[3]National Center for Toxicological Research, US Food and Drug Administration, Jefferson, AR 72079, USA
[4]Cancer Institute, Zhejiang University, Hangzhou, 310009, China
[5]Breast Medical Oncology, University of Texas M. D. Anderson Cancer Center, Houston TX, 77230 USA

Abstract— **Feature selection is one of the most important research topics in high dimensional array data analysis. We propose a two-way filtering based method that utilizes a pair of statistics coupled with rigorous cross-validation to identify the most informative features from different types of distributions. We evaluate the utility of the proposed adaptive feature selection method on six MicroArray Quality Control Phase II (MAQC-II) datasets. The results show that our method yields models with significantly fewer features and can achieve comparable or superior classification performance compared to models generated from other feature selection methods, suggesting high quality feature selection.**

Microarray data analysis; feature selection; predictive modeling; classifier learning; gene expression; biomarker discovery

I. INTRODUCTION

The goal of feature selection in high dimensional array data is to find informative genes (or any other molecular variables) from a set of examples with known clinical outcome. Feature selection serves two distinct purposes: (a) identify a parsimonious set of features (i.e. molecular variables or markers) that yield a predictive model with good performance in independent cases; (b) identify all significantly differentially expressed features between two outcome groups in order to gain insight into biological processes which differentiate the groups by making use of pathway analysis or gene ontology analysis tools.

There are three general techniques employed in feature selection [1] – (1) filter methods, which filters out unimportant features before the classifier learning process, (2) wrapper methods, which wrap a certain feature searching strategy around the classifier learning process, and (3) embedded methods, which utilize the internal feature selection ability of certain modeling techniques. In practice, microarray data analysis algorithms often combine multiple feature selection techniques to cope with high dimensionality. Many of these algorithms must perform feature ranking (scoring) using a certain statistical test either before or inside the classifier learning process.

Essentially most statistical tests for feature ranking are different measures of signal to noise ratio base on certain assumptions about the true distribution of the features. For features that follow the normal distribution a classical t-test is an excellent choice for measuring signal (mean difference between two phenotypes) to noise (variation) ratio. However, in array data analysis, many predictive features are far from being normally distributed and the noise cannot be effectively estimated since the sample sizes are usually small (an example of such a feature is shown in Fig. 1A). One solution is to focus on finding strong signals without much control of the noise. These statistics include mean difference (fold change) test, cancer outlier profile analysis (COPA) [2], outlier sum (OS) [3] and outlier robust *t*-statistic (ORT) [4] etc. Various modified t-tests, e.g., SAM [5], Efron-t (equation (2.8) of [6]), "shrinkage-t" [7], have also been proposed, primarily to balance the tradeoff between a mean difference test and a t-test in order to efficiently detect both normally distributed and non-normally distributed type of predictive features.

Since effective biomarkers may be distributed in any number of ways, and that different types of biomarkers may coexist in one dataset, it is impossible for a single statistical test to be optimal for all biomarkers. If a sub-optimal test is used, it may fail to detect certain informative features.

To address these issues, we have developed a two-way filtering feature selection method which selects features by searching for the desired thresholds of a pair of statistics that are used to filter features. For any pair of thresholds, the features that satisfy both thresholds are used to build a certain classifier. When choosing the pair of statistics, we choose one that is more efficient at detecting strong signals such as mean difference test, and a second that is more efficient at controlling signal to noise ratio such as classical t-test or Mann-Whitney U test. By varying the thresholds of these two statistics in certain steps within their acceptable ranges, we can achieve various tradeoffs and control the size of the feature sets.

We choose diagonal linear discriminate analysis (DLDA) classifier as the standard classifier for our features selection method because of its simplicity and effectiveness [8]. DLDA is one of the simplest linear classifiers similar to weighted voting [9] and it has no parameter to be tuned. Therefore, the pair of the thresholds of the two statistics solely determines the feature set and thus the DLDA model.

978-1-4673-2560-8/12/$31.00 ©2012 IEEE

For each combination of the two thresholds, useful metrics are calculated, such as cross-validated DLDA prediction performance, fold change and size of feature set. By collecting results from all pairs of thresholds, an overview analysis can be performed to identify where the optimal tradeoffs may exist (see Fig. 1B). In a sense, our two-way filtering method can be viewed as a generalization of various modified t-tests when the pair of statistics used is mean difference test and t-test. Instead of using a fixed tradeoff, the proposed adaptive method will automatically discover optimal tradeoffs for a given dataset.

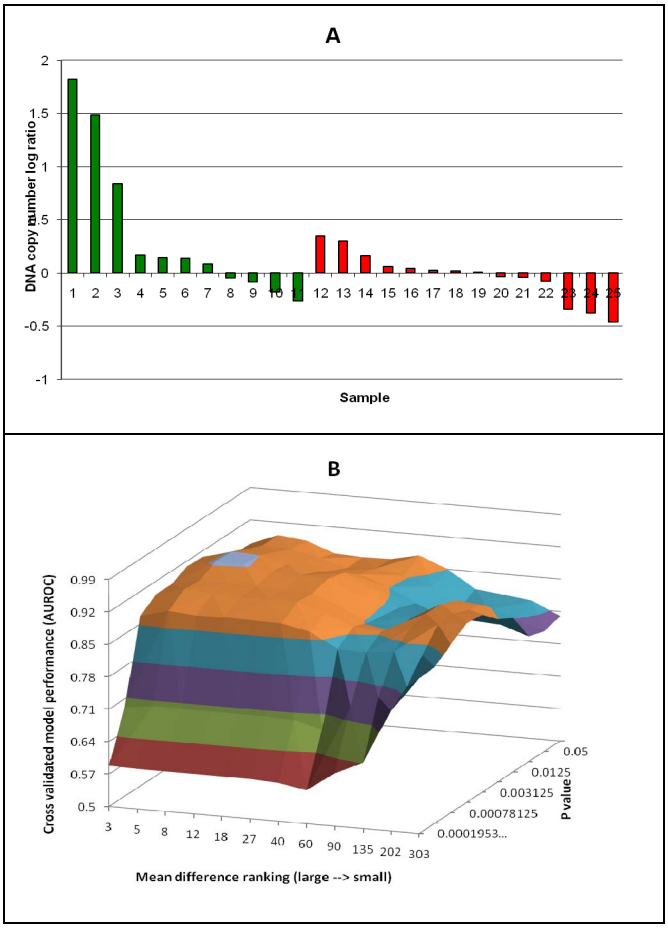

Figure 1. Discovering biomarkers for predicting lapatinib efficacy using baseline DNA copy number data of 25 cell lines. (A) An example of a non-nomrally distributed feature – HER2. Positive values indicate copy number gains. The green bars represent lapatinib sensitive cell lines and the red bars represent lapatinib resistant cell lines. High elevation in HER2 can only explain a small portion of the lapatinib sensitive cell lines. Because cancer is a highly heterogeneous disease, this type of marker is quite common in cancer studies. (B) The performance surface of our proposed two-way filtering feature selection. The pair of statistics used are mean difference test and t-test. Each point in the surface shows the cross validated DLDA model performance using features that satisfy both cutoffs. The top performing region near the upper left corner indicates that the optimal tradeoff for this dataset requires very stringent mean difference cutoff and less stringent p value cutoff (such as top 8 features by mean difference ranking and t-test p<0.0125). HER2 feature is included in all top models, which only have 3 to 5 features. As shown in the plot, more stringent p value cutoffs result in performance loss.

As different types of biomarker signals can coexist in the same data set, an iterative wrapper procedure was developed to enable the discovery of more features by finding a number of effective tradeoffs between the two statistics.

The proposed method and other supporting functions are implemented in a Java based tool called Array Data Analyzer (ADA), which has been applied to various GSK drug discovery projects [10]. It has also been applied to the FDA MAQC-II project [11] by the GSK data analysis team (DAT). The GSK DAT achieved highest mean area under the receiver operating characteristic curve (AUROCC) across all 11 endpoints among the participating DATs. The ADA tool allows users to choose different pairs of statistics to perform the two-way filtering. In this paper, the default pair of statistics is used, which are the mean difference test and the Mann-Whitney U test.

Experimental results on all six MAQC-II datasets show that models based on the proposed method have the smallest number of features on average while achieving comparable or superior prediction performance compared to those based on single statistic feature selection methods.

II. METHODS

A. *Main functions of the ADA tool*

Given a training dataset, the ADA tool may be used to answer three basic questions:

- *What is the parsimonious feature list which results in an optimal prediction model for a given data set?*
- *What level of performance should we expect for this prediction model in an independent validation?*
- *What is the expanded list of differentially expressed genes for biological pathway analysis?*

Three core functions of the ADA tool were developed to facilitate answering these questions.

The procedure 'findGeneSignature' helps answer the first question using a grid search procedure that searches through various tradeoffs of the pairs of statistics. When no user preference is given, the method will use the optimal balance (in terms of cross-validated model performance) of the two statistics to generate a prediction model. This is the same procedure used to generate the results for the proposed method when comparing feature selection methods (Figures 2 and 3). It was also used for the submission of prediction results to the MAQC-II project.

A nested cross-validation (CV) procedure called 'estimatePerformance' is used to address the second question. In this procedure, procedure 'findGeneSignature' is called to tune the parameters using the inner cross-validation. The outer cross-validation is used to estimate model performance.

The third question is analyzed using procedure 'findImportantGenes', which is a wrapper procedure that iteratively collects generated gene signatures and removes those genes from further runs. This process continues until procedure 'estimatePerformance' returns close to random performance (i.e., most informative genes have been identified and removed).

B. Pseudo code for main functions

High level pseudo code for the core functions is given below. The code is for illustration purpose only and is not optimized for efficiency. Statistics A and B can be any suitable statistical tests as long as one is primarily designed for detecting strong signals and the other for controlling signal to noise ratio. The tests used in this paper are the mean difference test and the Mann-Whitney U test. For computational efficiency, the procedure findGeneSignature only considers the top N features with the largest fold changes. When comparing feature selection methods, we set N equal to 300. This also ensures that all of the identified features have reasonably large fold changes.

```
Procedure trainModel (training data, a pair of
thresholds for statistic A and statistic B
respectively) {
    1. Collect statistic A and statistic B for
       each gene based on training data
    2. Using both statistics to filter genes
       according to the thresholds.
    3. Build DLDA model using the genes that
       passed the filtering process.
    4. Return DLDA model
}

Procedure testModel (test data, model) {
       Use the DLDA model to return continuous
scores for test cases
}

Procedure findGeneSignature (data, ranges and
steps of thresholds for statistic A and statistic
B) {
    1. For each combination of the two thresholds,
       call (repeated) cross-validation procedure
       to get averaged performance and averaged
       size of feature set. Procedure trainModel
       and procedure testModel are called
       repeatedly within the cross-validation
       procedure
    2. Select the optimal pair of thresholds based
       on model performance. (Users can also hand
       pick a pair of thresholds based on
       performance, size of feature size, and fold
       change etc.)
    3. Call procedure trainModel(training data,
       selected pair of thresholds)
}

Procedure estimatePerformance (data) {
    1. Run (repeated) outer cross-validation to
       estimate model performance. Procedure
       findGeneSignature is called repeatedly
       within the cross-validation procedure using
       outer CV training data. Models returned
       from procedure findGeneSignature are
       repeatedly evaluated by procedure testModel
       using outer CV test data. (The CV inside
       procedure findGeneSignature is referred to
       as inner CV.)
    2. Outer CV results are collected and the
       averaged performance is returned.
}

Procedure findImportantGenes (data, performance
threshold) {
    • While (procedure estimatePerformance (data)
      > performance threshold) {
        • Call procedure findGeneSignature
        • Add the genes of the returned model to
          the list of important genes
        • Remove these genes from the training
          data
      }
    • Return the list of important genes
}
```

C. Data preprocessing

The six MAQC-II datasets used in the experiment are the normalized datasets provided by MAQC-II project, which were processed following the standard procedure (see [11] for details). For all datasets generated from Affymetrix platforms, if a gene intensity value is smaller than 40, we change it to 40. The log transformed expression values are used for analysis.

III. RESULTS

A. MAQC-II datasets

MicroArray Quality Control (MAQC) project is an FDA led initiative to evaluate different practices for processing and analyzing microarray data. Phase II of this project focuses on evaluating different approaches for feature selection and predictive modeling using six datasets (13 endpoints) with blinded validation sets. Thirty six data analysis teams (DATs) from academia, industry and government agencies voluntarily submitted data analysis plans and subsequently prediction results for the blinded validation sets.

All six data sets of the MAQC-II project are used to empirically evaluate the proposed two-way-filtering method and nine other feature selection methods. For details of these datasets see the MAQC-II main paper [11]. As defined in the MAQC project, 11 of the 13 endpoints are used for performance evaluation and two negative control endpoints are ignored.

B. Comparing feature selection methods

We compare our approach against nine other simple feature selection techniques (see Table 1), each coupled with DLDA to generate an optimal model, which is then used to predict the cases in the blinded validation set. The average area under the receiver operating characteristic curve (AUROCC) performance of six data sets with 11 endpoints for each feature selection approach is shown in Fig. 2. The number of features to be included in a predictive model is optimized using a cross-validation technique and the search range is from 1 to 90 features. Since this is the same procedure to evaluate different DATs, we also included the performance of different DATs of MAQC project in Fig. 2. The result shows that all these simple feature selection methods performance extremely well when coupled with DLDA. Only top 3 DATs of MAQC-II achieved similar performance.

TABLE I. FEATURE SELECTION METHODS INCLUDED IN THE EXPERIMENT

Methods	Description
FC	fold change method (mean difference test for log transformed data)
FC+t	fold change test with t-test filter (feature with t-test p value > 0.05 are filtered out)
T	conventional t-test
T+fc	t-test with fold change filter (features with fold change <2 are filtered out)
Efron T	a modified t-test based on Efron's 90% rule (Efron et al, 2001)
Shrink T	a modified t-test based on shrinkage estimate of variance of each gene (Opgen-Rhein and Strimmer, 2007)
U	Mann-Whitney U test
U+fc	U test with fold change filter (features with fc<2 are filtered out)
Ensemble	always pick the model with the best cross-validation performance among the models from the above eight methods

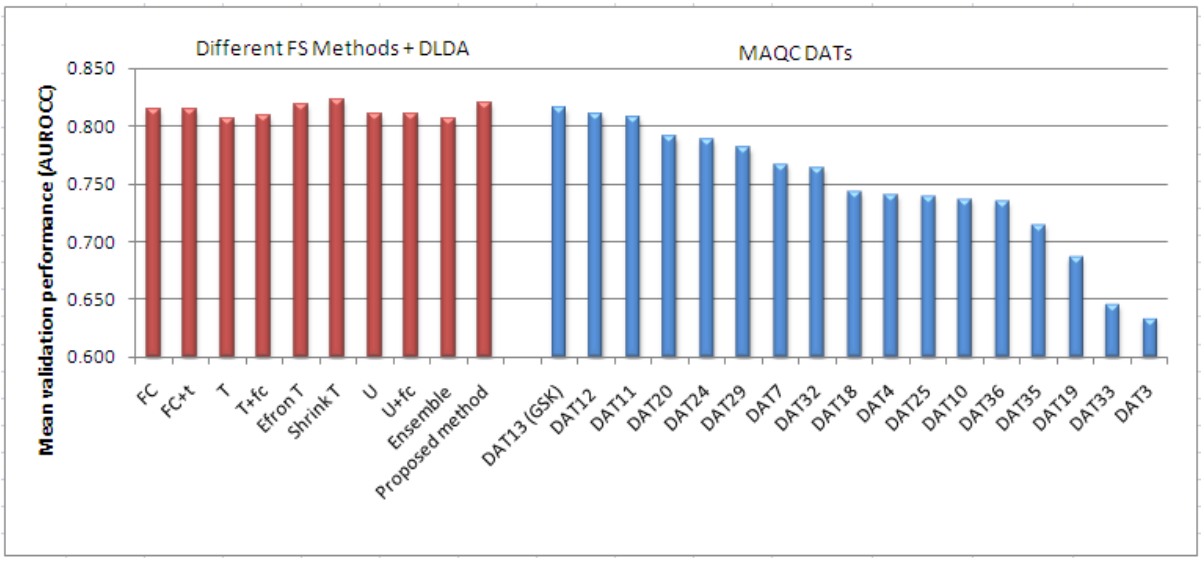

Figure 2. Mean validation performance in term of *AUROCC* of 11 MAQC endpoints for 10 feature selection methods coupled with *DLDA* modeling. The reported performance of 17 DATs of MAQC project is also included for reference. DAT13(GSK) uses a slightly different version of the proposed two-way filtering method coupled with *DLDA*.

Besides the DLDA model performance, we are also interested in evaluating the size of the DLDA model (number of features) for these feature selection methods since smaller models are often preferable in clinical settings. In order to study the variability of the model size, we performed bootstrapping by generating 50 new sets of the training data through random re-sampling of the original training data. Each new set of training data set is used to train optimal models and these models are evaluated using the independent validation set. The resulted mean model performance of the 50 bootstrap runs for each approach is very close to what we observe in Fig. 2 – all the approaches listed in Table 1 perform quite well. However, the mean model sizes of different approaches are quite different. Fig. 3 shows that our proposed feature selection yields much smaller models when compared to other approaches yet achieving similar or better performance, suggesting that it is more effective in picking out important features.

C. *Running time of the proposed method*

The proposed method is more computationally expensive than other feature selection methods listed in Table 1. However, the computation time is quite manageable and all experiments in this paper were conducted on a modest laptop PC (Dell Latitude E4300 with 2GB of RAM). For example, finding an optimal gene signature from microarray gene expression dataset with 130 samples takes about 50 seconds (grid search through 200 combinations of the threshold pairs of the two statistics; using 5 times 5 fold CV to measure performance of each combination). If we would like to get unbiased model performance estimation using nested CV (5 times 5-fold outer CV and 5 times 5 -fold inner CV), it would take about 20 minutes.

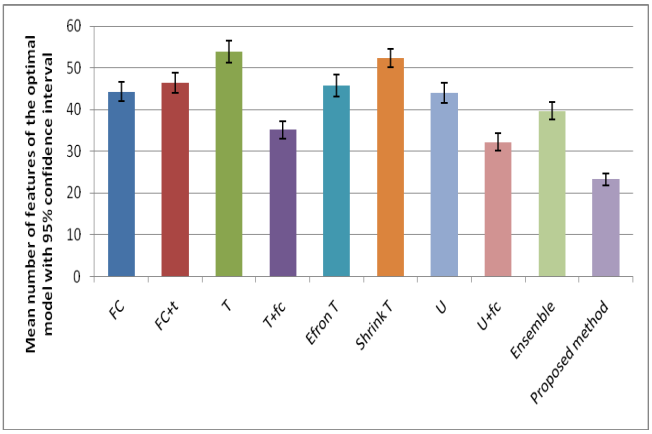

Figure 3. This figure shows the mean DLDA model size (number of features) of 50 bootstrap runs using the same 11 endpoints, with the error bars indicating the 95% confidence intervals.

IV. Discussion

In Fig. 2, we use *DLDA* model performance to evaluate the effectiveness of different feature selection methods. Although model performance is an important measure for this purpose, there are other measures that are also important in real world applications, such as the average fold change and the size of a feature set. In practice, researchers at times are willing to sacrifice performance in order to gain other logistic properties, such as smaller numbers of features and larger fold change. These properties enable more tractable assays for clinical use (e.g. qPCR). The proposed method is a convenient tool to facilitate this requirement. By visually examining the performance at different combinations of cut points of the two statistics (See Fig. 1B) and checking the size of feature sets, researchers can decide which feature set would be best to use. Fig. 3 shows that the proposed method has the smallest feature sets on average, which is not surprising as the two-way filtering provides more flexibility in finding high performing sets of features parsimoniously.

Fig. 2 shows that all feature selection methods being evaluated (, when coupled with DLDA modeling,) can achieve similar or better performance than the top DATs of MAQC-II. Although it is not completely valid to compare these results to the blinded test results of MAQC-II DATs, it still suggests that DLDA is a very effective modeling tool for high dimensional array data, especially considering the fact that DLDA itself needs no training and no parameters to tune. As a general feature selection framework, the DLDA classifier can be easily replaced by other classifiers in our ADA tool. However, we doubt that other classifiers can significantly improve the prediction performance.

Besides evaluating our proposed method and eight feature selection methods based on single statistics, we also included an "ensemble" method in our experiment, which always picks the best model based on cross-validation performance among the models generated by the eight feature selection methods. Although this kind of learning scheme is quite popular, the results show no evidence that the "ensemble" method was more effective than standard feature selection methods. One explanation is that by selecting the best model from many different models, the "ensemble" method actually increases the variance of the modeling procedure, rather than reducing the bias.

Our approach relies heavily on robust cross validation to control over fitting. We believe that this is an efficient way to use the relatively small number of samples – artificially defining a small validation set is unlikely to achieve much due to the large test data variability and reduced power of model development in the smaller training set. However, it is crucial that the cross-validation is done properly. For example, the feature selection must be performed within each run of cross validation; and the nested cross validation is often required when evaluating model performance (i.e., outer loop CV for model performance evaluation and inner loop CV for model parameter tuning). As we can see here, the proper cross-validation procedures can often be quite computationally expensive and may require thousands of times of model training and testing. This is the most important reason that a simple modeling technique such as DLDA is preferred. We believe the best way to gain better performance is through improving performance of feature selection, rather than tuning modeling parameters of complex models. Complicated learning schemes can make proper cross-validation too computational expensive to run. Without proper cross-validation, the result can be overly optimistic.

We use the ADA tool for both feature selection and predictive modeling. However, the tool can also be used for feature selection only and the selected features can then be used to train a more sophisticated classifier such as a support vector machine.

Acknowledgment

The authors would like to thank Dr. Pankaj Agawal of GlaxoSmithKline for his valuable comments.

References

[1] Saeys Y, *et al.* (2007) A review of feature selection techniques in bioinformatics. Bioinformatics; 23:2507-2517.

[2] Tomlins SA, *et al.* (2005) Recurrent fusion of TMPRSS2 and ETS transcription factor genes in prostate cancer. *Science* 310: 644–8.

[3] Tibshirani,R. and Hastie,T. (2007) Outlier sums for differential gene expression analysis. *Biostatistics*, **8**, 2–8.

[4] Wu,B. (2007) Cancer outlier differential gene expression detection. *Biostatistics*, **8**,566–575.

[5] Tusher VG, Tibshirani R, Chu G: (2001) Significance analysis of microarrays applied to the ionizing radiation response. *Proc Natl Acad Sci USA* 98(9):5116-5121.

[6] Efron, B., Tibshirani, R., Storey, J. D., and Tusher, V. (2001), "Empirical Bayes Analysis of a Microarray Experiment," Journal of the American Statistical Society, 96 (456), 1151–1160

[7] Opgen-Rhein, Rainer and Strimmer, Korbinian (2007) "Accurate Ranking of Differentially Expressed Genes by a Distribution-Free Shrinkage Approach," Statistical Applications in Genetics and Molecular Biology: Vol. 6: Iss. 1, Article 9.

[8] Dudoit R, Fridly J, Speed TP. (2002) Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data. *JAMA* Vol. 97 No.457, 77-87.

[9] Golub, *et al.* (1999) Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science Vol 286, Oct 1999.

[10] Greshock, J. *et al.* (2008) Genome-wide DNA copy number predictors of lapatinib sensitivity in tumor-derived cell lines. *Mol. Cancer Ther.* **7**, 935-943

[11] MAQC consortium (2010), The MicroArray Quality Control (MAQC)-II study of common practices for the development and validation. *Nat Biotechnol* 28, 827-838.