A Model-Free Greedy Gene Selection for Microarray Sample Class Prediction

Yi Shi*, Zhipeng Cai*
Department of Computing Science, University of Alberta
Edmonton, Alberta T6G 2E8, Canada
Email: ys3, zhipeng@cs.ualberta.ca

Lizhe Xu
The Research Institute for Children-University of New Orleans
New Orleans, LA 70118, USA
Email: lizhe.xu@dhs.gov

Wei Ren
Academy of Mathematics and Systems Science, Chinese Academy of Sciences
Beijing 100080, China
Email: renwei@amss.ac.cn

Randy Goebel, Guohui Lin†
Department of Computing Science, University of Alberta
Edmonton, Alberta T6G 2E8, Canada
Email: goebel, ghlin@cs.ualberta.ca

***Abstract*— Microarray data analysis is notoriously challenging as it involves a huge number of genes compared to only a limited number of samples. Gene selection, to detect the most significantly differentially expressed genes under different categories of conditions, is both computationally and biologically interesting, and has become a central research focus in all studies that use gene expression microarray technology. Despite many existing efforts, better gene selection methods that can effectively identify biologically significant biomarkers, yet computationally efficient, are still in need. In this paper, a model-free greedy (MFG) gene selection method is proposed, which implements several intuitive heuristics but doesn't assume any statistical distribution on the expression data. The experimental results on three real microarray datasets showed that the MFG method combined with a Support Vector Machine (SVM) classifier or a k-Nearest Neighbor (KNN) classifier is efficient and robust in identifying discriminatory genes.**

***Index Terms*— Microarray data analysis, sample class prediction, discriminatory gene, gene selection, greedy.**

I. Introduction

The fast developing microarray technology allows us to monitor the expression levels for thousands of genes simultaneously. This novel technique provides us a large amount of data to understand systematically various gene regulations under different conditions as well as their relation to diseases and treatments. Nevertheless, though microarray technology expands biological and medical information tremendously, it also brings the challenges of how to use the data reasonably and thus to abstract useful information. With respect to a specific application, we may have two categories of genes. For one category of genes, their expression levels are largely unchanged under different conditions, such as house-keeping genes. These genes are less interesting since they do not provide useful information related to the samples. The second category of genes are those that are differentially regulated under certain experimental conditions, that is, their expression levels either increase or decrease across different conditions. These *discriminatory* genes are very important in delivering information related to experimental conditions and the samples' property. For instance, they could be highly correlated to certain kinds of diseases or medical conditions. Gene selection is about looking for the second category genes.

Since discriminatory genes are differentially regulated under certain experimental conditions, or in other words they are expected to have very close expression levels in samples of the same condition, or *class* called in this paper for the sample classification purpose, but significantly different across samples from different classes, there are many gene selection methods proposed to capture this intuitive criterion, for example the F-test [1], [2] and its variants [3], [4]. Basically, these gene selection methods implement the criterion to assign each gene a score and then sort the genes into a non-increasing order, where a higher score indicates a more differential expression [1], [2], [3], [4]. According to Liu and Yu [5],

*YS and ZC contributed equally to this work.
†To whom correspondences should be addressed.
1-4244-0623-4/06/$20.00

they belong to the *wrapper-sequential-classification* category of feature selection methods. They then set up a score cut-off to report those genes having scores higher than the cut-off. The quality of the subset of reported genes is measured by their class prediction power, when combined with a classifier such as a support vector machine (SVM) classifier or a k nearest neighbor (KNN) classifier, through a leave-one-out cross validation (LOOCV) or an ℓ-fold cross validation.

Microarray data classification is one of the most computationally challenging tasks, which includes previously unrecognized class discovery and sample class prediction. With known classes, the main purpose of gene selection is to identify those discriminatory genes whose expression levels signify the class. In this sense, the classification accuracy of the identified gene subset indeed reflects their biological significance as a whole. The above mentioned gene selection methods only report a certain number of top ranked genes, without consideration of the facts that some genes may have very similar discrimination power as individuals and some genes may have complementary discrimination powers. Obviously, for genes having very similar discrimination powers, if one is top ranked, then the others could also be top ranked. Consequently, using them all in class prediction is redundant. On the other hand, for genes having complementary discrimination powers, they might not be top ranked since as individuals their discrimination powers could be lower than the cut-off. As a result, they would not be used for class prediction and it leads to a loss that could not be made up by the use of other top ranked genes. In this paper, we propose a gene selection method to partially address the above two issues to select a subset of genes that have superior discrimination power as a whole. According to the categorizing framework by Liu and Yu [5], our method might still belong to *wrapper-sequential-classification*, yet it takes some idea from *filter* algorithms to exclude or remove certain genes that do not contribute much to the existing selected gene subset.

We note that there are a few previously proposed methods that attempt to address the same issues, to be reviewed in the Methods section. Nevertheless, compared to them, our method implements heuristics that are intuitively sound and is computationally very efficient. For example, our method takes only seconds to minutes on a normal size real microarray dataset. Our method is also *model-free* in that it does not assume or require any statistical model on the gene expression levels. We review some of the recent related work in the following.

There are a variety of gene selection methods that have been proposed recently, among the rich literature of feature selection algorithms [5]. In general, gene selection methods can be partitioned into two categories, which are model-free and model-based. Model-based gene selection methods assume some specific statistical models on the gene expression data. For example, Baldi *et al.* [6] developed a Gaussian gene-independent model to process the gene expression data. They implemented a t-test combined with a full Bayesian treatment on the gene expression data. Obviously, the disadvantage of this category of gene selection methods is the lack of adaptability, because it is unlikely to construct a universal probabilistic analysis model that is suitable for all kinds of gene expression data, where noise and variance vary dramatically across different gene expression datasets [7]. Model-free gene selection methods do not assume any specific distribution model on the gene expression data. For example, Xiong *et al.* [8] suggested two methods, sequential forward selection (SFS) and sequential forward floating selection (SFFS), to select genes through the space of gene subsets using the classification error. We will compare our method with SFS and SFFS, whose more detailed descriptions are included in the next section. Guyon *et al.* [9] proposed a gene selection approach utilizing support vector machines based on recursive feature elimination (RFE). These model-free gene selection methods, however, have been reported [7] to be possibly influenced by the specific criteria used for scoring the gene discrimination power. According to the categorizing framework by Liu and Yu [5], most of these methods belong to *filter*/*wrapper-sequential-classification* category.

The gene selection method proposed in this paper is model-free and implements three greedy heuristics on discrimination power, called the MFG method. The MFG method does not assume statistical model on the gene expression levels, and it adopts the classification accuracy of an individual gene as the score. We have tested two classifiers, a linear kernel support vector machine (SVM) classifier and a k nearest neighbor (KNN) classifier for $k = 5$, to combine with the MFG method for measuring the classification accuracy. After each gene is assigned a score, they are sorted in non-increasing order and the selection starts with the gene of the highest score. Three heuristics implemented in the selection process are *greedy*, *c-kick* and *exchange*, to be detailed in the Methods section. The MFG method is compared with four recently proposed methods, SFS, SFFS, Cho's and F-test, on eight real microarray datasets, among which we chose to present results on three datasets in the current report. Note that gene selection is essentially designed to reduce the dimensionality of the gene space, since there are usually much more genes than the number of the microarray chips. To demonstrate the effectiveness of reported genes, we also tested a method to randomly pick the same number of genes from the gene pool and examined their classification accuracy as a whole. We denote this method as *Random*, whose performance can be regarded as a baseline. The experimental results showed that the MFG method is very efficient and effective in identifying biologically meaningful genes that can be used for class prediction purpose.

The rest of the paper is organized as follows: In the next section, we introduce further details on gene selection and then present the details of the MFG method, F-test, Cho's, SFS and SFFS. Section III summarizes the experimental results of the MFG method combined with two classifiers, a linear kernel SVM-classifier and a KNN-classifier, on three real microarray datasets that are used for cancer subtype determination. We discuss the results in Section IV on different aspects of the MFG method and their effects. Section V concludes the paper and points out some immediate future works.

II. METHODS

Assume in each microarray chip there are in total n genes and in total m chips/samples in the dataset that have been grouped into L classes. In our tested datasets, there are usually multiple classes, typically in those three chosen to be reported. Therefore, the microarray dataset can be represented as a matrix of expression levels $A_{n\times m} = (a_{ij})_{n\times m}$, where a_{ij} denotes the expression level of gene i in sample j. Note that every sample is labeled with its class name in the original dataset.

Given a gene selection method, in order to test its performance, the microarray dataset is randomly partitioned into ℓ equal parts. Among them, $\ell - 1$ parts are used by the gene selection method to selected a number of discriminatory genes. These selected genes are then fed to a classifier, which is tested on the last part to see how well the classifier can tell the class membership for each sample (whose original class label is erased before the testing). Consequently, those $\ell - 1$ parts form a *training dataset* and the last part forms a *testing dataset*. The process is repeated ℓ times to have every part as the testing dataset and the average classification accuracy over these ℓ ones is called the *ℓ-fold cross validation classification accuracy* of the gene selection method combined with the classifier. In this current work, we chose $\ell = 5$ and the random partitioning process was repeated for 20 times. That is, the ℓ-fold cross validation classification accuracy is the average over a total of 100 ones.

We adopted two classifiers in our study, one is a linear kernel SVM-classifier [9] and the other is a KNN-classifier [1]. Essentially, SVMs compute a decision plane to separate the set of samples (in the training dataset) having different class memberships, and use this plane to predict the class memberships for testing chips. A KNN classifier, in our case $k = 5$, ascertains the class for a testing sample by analyzing its k nearest neighbors in the training dataset and by a majority vote. The interested readers might refer to [9], [1] for more details.

In the following, we present the MFG method on the training dataset in details, with the understanding that this stage only returns a subset of genes. How to build a classifier using these reported genes and the subsequent testing follow the above description and the separate classifiers [9], [1].

A. The MFG Method

In the training dataset, the expression levels of all the samples and their class labels are used for gene selection purpose. In the MFG method, firstly, each gene is used to build a classifier (which could be the SVM-classifier or the KNN-classifier). Adopting the 5-fold cross validation scheme, the performance of the classifier measured by the classification accuracy is the score assigned to the individual gene. In other words, we adopt the cross validation classification accuracy of an individual gene as the *scoring function* to rank genes. Such a scoring function does not assume any statistical model on the expression data (but the adopted classifier) and thus the MFG method is model-free. We remark that in this stage of scoring genes, the training dataset itself is partitioned into 5 parts for 5-fold cross validation purpose, and the gene scoring has nothing to do with the testing dataset. After each gene is assigned a score, they are sorted in the non-increasing order.

The optimization problem in gene selection is to select a subset of genes that have the highest classification accuracy as a whole, which, however, turns out to be NP-hard. Several heuristics can be applied to rapidly identify a subset of genes that might not have the highest classification accuracy but intuitively very close to the highest. Note that using more genes might not always be a better choice, and thus the set of all genes is generally not the desired solution. One of the simplest heuristics is probably to return the subset of top ranked x genes in the sorted gene order, where x is a number specified by the user. Another simplest heuristics is to randomly select a subset of x genes, even without using the sorted gene order. We denote this latter method as *Random*, whose performance might be regarded as a baseline of the dataset. That is, every good gene selection algorithm should perform better than Random in order to be recommended.

The Random method certainly is a blind search, without taking advantage of individual gene information. On the other hand, simply reporting the top ranked a few genes does not address the aforementioned two issues: 1) Some similarly expressed genes might all be top ranked and using them all in the classification is redundant; 2) Some genes having complementary discrimination powers might not be top ranked as individuals and thus not included for building classifiers. The proposed MFG method intends to resolve these two issues, through implementation of the following three heuristics. It essentially scans through the sorted gene order to pick up genes which it believes useful, and once every a few iterations it removes some genes which it believes not useful any more.

1) The Heuristics: There are three heuristics implemented in the MFG method:

a) Greedy: The *greedy* heuristics allows the MFG method to include a gene only when the gene under consideration can improve the classification accuracy, upon appending it to the current selected gene pool. In more details, assume the MFG method is examining the i-th gene g_i in the sorted order, and the current selected gene pool is P, which is a subset of the first $i - 1$ genes. Let q be the classification accuracy of gene set P as a whole; let q' be the classification accuracy of gene set $P+g_i$. If $q' > q$, then gene g_i is added to the selected gene pool and the MFG method either proceeds to consider the $(i + 1)$-st gene in the sorted gene order, or applies the following *c-kick* heuristics. If $q' \leq q$, then gene g_i is regarded as useless with respect to P and the MFG method proceeds to apply the *exchange* heuristics.

b) Exchange: At one iteration of the MFG method where the gene under consideration doesn't improve the classification accuracy, MFG does not discard the gene immediately. Instead, it implements the following heuristics to examine one step backward to see whether or not this gene indeed can be discarded. To this purpose, again assume the notations in the last paragraph and let g_j denote the last gene added to the current selected gene pool P. Note that the individual discrimination power of gene g_j is at least as high as that of gene g_i. The MFG method tests the classification accuracy

q' of gene set $P-g_j+g_i$, that is, to replace the last added gene g_j by gene g_i. If $q' \leq q$, then gene g_i is discarded from further consideration. Otherwise, the replacement is taken and gene g_j is discarded from further consideration. In either case, the MFG method proceeds to consider the next gene in the sorted gene order. Such an *exchange* heuristics is designed to pick up genes that have complementary discrimination powers to some already selected genes.

c) c-Kick: Once every a few iterations, the MFG method re-examines its selected gene pool and tries to remove some of them that do not contribute to the overall classification. This *c-kick* heuristics is different from the exchange heuristics that only considers to replace the lastly added gene by the current gene. Rather, in more details, once the MFG method has continuously added c fresh genes, say $g_1, g_2, \ldots, g_c$ in the order of addition, it scans if kicking any one of them out of the selected gene pool will increase the classification accuracy. If kicking out one gene does increase the classification accuracy, then the gene is removed from the selected gene pool and would never be considered again. The MFG method continues till no more gene can be kicked out, and then resumes the scanning of the sorted gene order. Inside this heuristics, c is a parameter specified by the user. We have tested several values for c and the best performance seemed achieved at 5. Note that both the *c-kick* and the *exchange* heuristics take into account a certain portion of the mutual information between genes, and search for the local optimal combinations. In this sense, implementing them into the MFG method expects better performance.

2) The Complete Description: To summarize, upon decision on a classifier and a cross validation scheme, the MFG method works on the training dataset to sort the genes in non-increasing order of their individual classification accuracy and then select a number of top ranked genes according to the three heuristics. These genes are used to build a classifier that can predict the class memberships for the samples in the testing dataset. The cross validation classification accuracy on the testing datasets is taken to measure the performance of the MFG method, combined with the chosen classifier, on the microarray dataset. Figure 1 contains a high level description of the MFG method.

INPUT: A microarray training dataset, a classifier, a cross validation scheme, c and x.
OUTPUT: A subset of x genes.
1) Apply the classifier to determine the cross validation classification accuracies of individual genes on the training dataset, and sort the genes in non-increasing order;
2) For each gene in the sorted order
   a) If it improves the classification accuracy
      i) Add it to the selected gene pool;
      ii) Execute *c-kick* heuristics if applicable;
   Else
      i) Execute *exchange* heuristics if applicable;
   b) If there are x genes selected or all genes have been considered
      i) Stop and return the selected genes;

Fig. 1
A HIGH LEVEL DESCRIPTION OF THE MFG METHOD.

B. The Other Methods

Feature selection is a process to select only a subset of original features whose optimality is measured by an evaluation criterion. Such a process is in general intractable and many related problems have been shown to be NP-hard. Liu and Yu [5] surveyed feature selection algorithms for classification and clustering, and proposed a two dimensional categorizing framework for the algorithms for classification. Most of the gene selection algorithms in the microarray data analysis literature for sample class prediction belong to the *wrapper-sequential* category, in particular the following four algorithms with which the MFG method is compared in this work.

1) F-test: In F-test, genes that have small intra-class variances and large inter-class variances will be ranked high. Formally, for each gene g_i, let $\overline{a}_j$ denote the mean expression value in samples in the j-th class and $\overline{a}$ denote the mean expression value in all the samples. The score of this gene is calculated as

$$\frac{\sum_{j=1}^{L}(\overline{a}_j - \overline{a})}{\sum_{j=1}^{L}\sum_{i=1}^{n_j}(a_{ij} - \overline{a}_j)},$$

where n_j is the number of samples in the j-th class. The first x top ranked genes are returned as selected genes. We note that though F-test is often reported inferior, it is a classic method to be compared with.

2) Cho's: Using the same notations used as in the above, Cho's method [3] defines a weight factor w_j for sample j, which is $\frac{1}{n_k}$ if sample j belongs to class k, whose size is n_k. Let $W = \sum_{j=1}^{m} w_j$. The weighted mean for gene i, denoted as $\tilde{a}_i$, is defined as

$$\tilde{a}_i = \sum_{j=1}^{m} \frac{w_j}{W} a_{ij}.$$

The weighted standard deviation, denoted as $\tilde{\sigma}_i$, is defined as

$$\tilde{\sigma}_i = \sqrt{\frac{m\sum_{j=1}^{m}(a_{ij} - \tilde{a}_i)^2}{(m-1)\sum_{j=1}^{m} w_j}}.$$

Then the score of gene i is calculated as

$$\frac{\tilde{a}_i \times \tilde{\sigma}_i}{\sigma},$$

where σ is the standard deviation of class centroid expression values for gene i: $(\overline{a}_{i1}, \overline{a}_{i2}, \ldots, \overline{a}_{iL})$, where $\overline{a}_{ik} = \frac{1}{n_k}\sum_{j=1}^{n_k} a_{ij}$. Likewise, the first x top ranked genes are returned as selected genes.

3) SFS: The *sequential forward search* (SFS) method has been proposed for general feature extraction a long time ago [10], but was only recently investigated for microarray data analysis [8]. Similarly as in the MFG method, each gene is assigned a score indicating its discrimination power — the classification accuracy. The top ranked gene is then selected. Next, this selected gene is combined with every other gene to determine the combination of two genes that achieves the highest classification accuracy. Then, this combination of two genes is combined with every other gene to determine the combination of three genes that achieves the highest classification accuracy, and so on. The process stops when a pre-specified number of genes have been selected, or there is no further improvement on the classification accuracy.

4) SFFS: Adding one more feature called *floating* to SFS, the *sequential forward floating search* (SFFS) method was also investigated for microarray data analysis [10], [8]. This was designed to overcome the so-called *nesting* effect in the SFS method, that is, once a gene is selected, there is no way for it to be discarded later on. In this sense, it is very similar to the c-kick heuristics in the MFG method. SFFS maintains a sequence of subsets of selected genes, which contain $1, 2, 3, \ldots, n$ genes, respectively. The subset of k selected genes is the one that to the moment achieves the highest classification accuracy using k genes. Applying SFS, assume that at the current iteration gene g_i is the one that achieves the highest classification accuracy when added to the current selected gene pool P. SFFS examines if there is one gene, say g_r, in P such that $P - g_r + g_i$ achieves a higher classification accuracy than P. If there is no such gene g_r, then SFFS adds g_i to P and moves on to the next iteration. Otherwise, it updates P to be $P - g_r + g_i$, i.e. $P \leftarrow P - g_r + g_i$, and continues on to examine if there is one gene, denoted as g_s, in $P - g_i$ such that $P - g_s$ achieves a higher classification accuracy than $P - g_i$, and so on.

III. Experimental Results

A. Overview

We have compared the MFG method with four other gene selection methods, namely F-test [1], Cho's [3], SFS and SFFS [8] on eight real microarray datasets. We have also used the performance of the Random method as the baseline for comparison. Among these eight datasets, we chose to report the three most difficult datasets, where the difficulty of a dataset is measured by the number of genes versus the number of samples in the dataset, as well as the degree of unbalanced numbers of samples in the classes. As shown in the following, in general, the classification accuracy of the Random method also hints the difficulty of the dataset.

We adopted two classifiers built on the selected genes, one is a linear kernel support vector machines (SVM) classifier [9] and the other is a k nearest neighbor (KNN) classifier [1], where $k = 5$. The SVM we used in MATLAB is from `http://theoval.sys.uea.ac.uk/~gcc/svm/toolbox/` and the KNN is coded in MATLAB by ourselves. All the experiments were conducted in MATLAB environment (`http://www.mathworks.com`) on a cluster of 2.33GHz CPUs. We adopted the 5-fold cross validation to test the classification accuracy of these gene selection methods.

B. Dataset Descriptions

We describe next the three microarray datasets on which the results are reported. Descriptions of the other five datasets are available upon request, as well as the datasets.

The GLIOMA dataset [11] contains in total 50 samples in four classes, *classic glioblastoma*, *nonclassic glioblastoma*, *classic anaplastic oligodendroglioma* and *nonclassic anaplastic oligodendroglioma*. This dataset is from Affymetrix U95Av2 GeneChips. There are 14, 14, 7 and 15 samples in these classes, respectively. Each sample originally had $12,625$ genes. We adopted a standard filtering method [11] to remove genes with minimal variations across the samples. In more details, for this dataset, the expression intensity thresholds were set at 20 and 16,000 units. That is, all hybridization intensity values less than 20, including negative hybridization intensity values, were raised to 20; and those higher than 16,000 were shifted to 16,000. Genes, whose variation of expression values is less than 100 in difference or less than 3 in fold change between any two samples, were excluded. After this preprocessing, we obtained a dataset with 50 samples on $4,433$ genes.

The LUNG dataset [12] contains in total 203 samples in five classes, *adenocarcinomas*, *squamous cell lung carcinomas*, *pulmonary carcinoids*, *small-cell lung carcinomas* and *normal lung*. This dataset is from Affymetrix U95A GeneChips. There are 139, 21, 20, 6 and 17 samples in these classes, respectively. Note that this dataset is extremely unbalanced, as one class contains 20 times samples more than another (139 versus 6). Each sample originally had $12,600$ genes. Similarly, those genes with standard deviations less than 50 expression units were removed so that we obtained a dataset with 203 samples on $3,312$ genes.

The SRBCT dataset [13] contains in total 83 samples in four classes, *the Ewing family of tumors*, *Burkitt lymphoma*, *neuroblastoma* and *rhabdomyosarcoma*. (Note that we excluded 5 misclassified samples from the original dataset.) This dataset is from cDNA chips and no preprocessing was done to it. Every sample in this dataset contains $2,308$ gene expression values. There are 29, 11, 18 and 25 samples in these four classes, respectively.

C. Classification Accuracies

Given a subset of selected genes, building a classifier and testing its classification accuracy together form a round of experiment. It worths pointing out that the MFG, F-test and Cho's gene selection methods all take $O(n)$ rounds, where n is the number of genes in the microarray dataset. But SFS and SFFS methods take $O(n^2)$ and $O(mn^2)$ rounds, respectively, where m is the number of samples in the dataset, which are considerably more expensive. In our experiments, we have found that running SFS and SFFS on the three datasets all took more than two weeks without completion (the estimated running time was months to years), while the others only took minutes to a few hours. Consequently, on

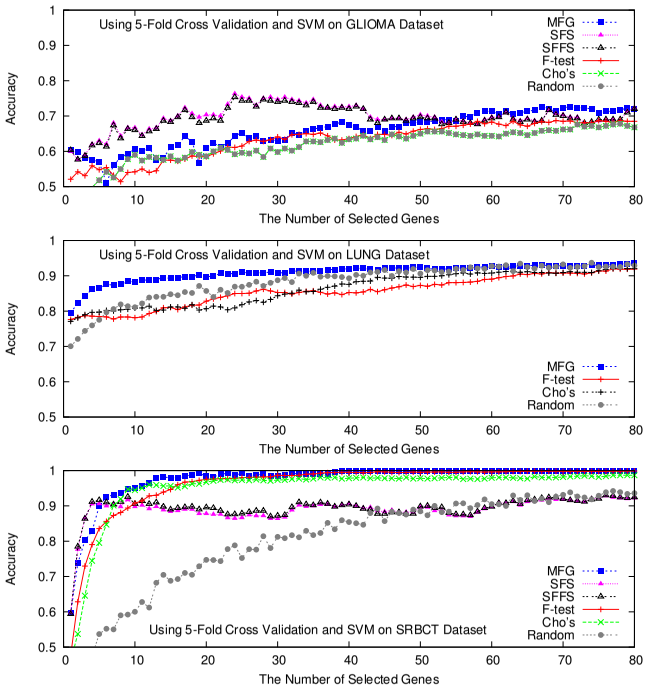

Fig. 2

SVM-CLASSIFIER 5-FOLD CROSS VALIDATION CLASSIFICATION ACCURACIES FOR ALL FIVE GENE SELECTION METHODS ON THE REDUCED GLIOMA DATASET, THE REDUCED LUNG DATASET, AND THE REDUCED SRBCT DATASET, RESPECTIVELY.

these three datasets, we only compared the MFG methods with F-test and Cho's gene selection methods. In order to compare with SFS and SFFS, we filtered further a lot more genes to obtain the *reduced* datasets, in which only $1,000$ the most differentially expressed genes were kept. We remark that such a filtering could remove informative genes from consideration. Nonetheless, since SFS and SFFS both adopt the same gene scoring function as in the MFG method, we believe that comparison must be done between the MFG method and SFS and SFFS, and this filtering process becomes essential to the comparison.

In all experiments, a gene selection method is always combined with a classifier. For simplicity, for example, the MFG-SVM classification accuracy refers to the 5-fold cross validation classification accuracy of the SVM-classifier built on the selected genes by the MFG method.

1) On Reduced Datasets: Note again that experiments on these reduced datasets are for the purpose of comparison among all five gene selection methods, plus the Random method. Figure 2 plots the SVM-classifier 5-fold classification accuracies for the gene selection methods MFG, F-test, Cho's, SFS and SFFS. The classification accuracy of the SVM-classifier built on the randomly selected genes is also plotted. In the plots, the x axis represents the number of selected genes and the y axis represents the classification accuracy.

From the plots in Figure 2, one can see that the reduced LUNG dataset is probably the hardest one among the three reduced datasets, in terms of identifying the discriminatory genes, since neither SFS nor SFFS can finish in two weeks (and thus no plots for them). Between the reduced GLIOMA dataset and the reduced SRBCT dataset, the plots of the Random-SVM indicate that the data quality of the reduced GLIOMA dataset could be lower, as every other gene selection method could not do much about selecting a gene subset to achieve high classification accuracies. The reduced SRBCT dataset is, however, a different story. On one hand, the classification accuracy of the Random method is significantly lower than all the others, typically than the MFG method. On the other hand, it indicates that the MFG method (and F-test) is much more effective than the SFS and SFFS methods. The experimental results on the full datasets in the next subsection also support the above conclusions.

2) On Full Datasets: On the full datasets, we could not obtain any results for SFS and SFFS on any one of the three datasets within two weeks. Therefore, in Figure 3, only the SVM-classifier 5-fold cross validation classification accuracies for the MFG method, F-test and Cho's are plotted. As expected, all the classification accuracies on the GLIOMA dataset and the LUNG dataset are crowded together, with those of F-test and Cho's intertwine with that of the Random method. This fact tells that essentially, on these two datasets, F-test and Cho's do not really select genes that are better than the Random method does. However, still, we might be able to see that the classification accuracies of the MFG method always stay higher than those of F-test, Cho's, and the Random method. In certain conditions, these differences were significant. Therefore, we might still be able to conclude that the genes selected by the MFG method have higher discrimination powers than the average. Also as expected, the experimental results on the SRBCT dataset clearly show that the genes selected by each of the three methods, MFG, F-test and Cho's, are better than those randomly selected, in terms of classification accuracy of the selected gene set as a whole. One can see that the MFG method does not outperform F-test and Cho's, which could be due to the good quality of the data — there is no room for further improvement. (In fact, many methods have been reported to perform equally well on this SRBCT dataset.)

Figure 4 plots the KNN-classifier 5-fold cross validation classification accuracies for the MFG method, F-test and Cho's on the three datasets GLIOMA, LUNG and SRBCT. Again, SFS and SFFS were not able to finish within a two week period of time and thus no results reported. All these plots support the conclusion that the MFG method performed at least as well as, and most of the time better than, the other two gene selection methods F-test and Cho's. SFS and SFFS could perform differently or better, but they took too long to have their results reported. Another observation from these plots is that, except a few cases on the GLIOMA dataset, in general the KNN-classifier performed worse than the SVM-classifier, which is particularly obvious on the SRBCT dataset. Lastly, from the performance comparison on the LUNG and the SRBCT datasets, the MFG method might not be a method that should combine with the KNN-classifier, though it performed extremely well (even better than the SVM-classifier) on the GLIOMA dataset.

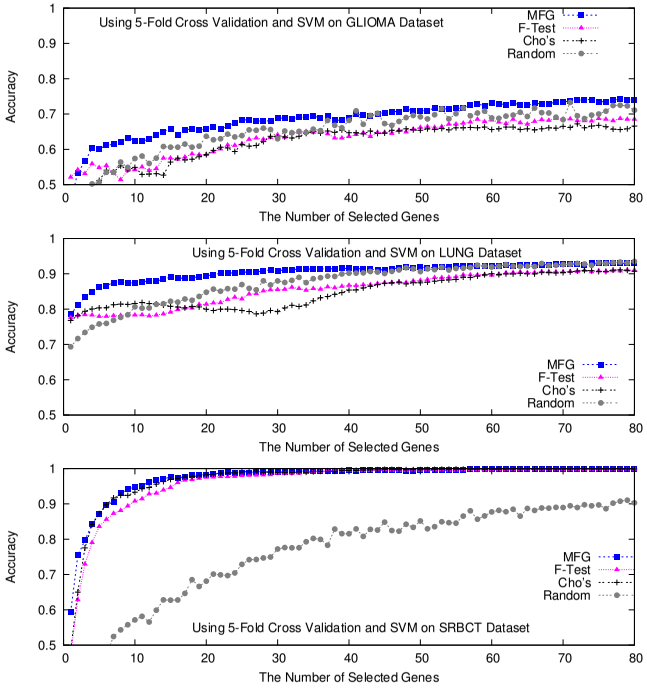

Fig. 3

SVM-CLASSIFIER 5-FOLD CROSS VALIDATION CLASSIFICATION ACCURACIES FOR THE THREE GENE SELECTION METHODS ON THE GLIOMA DATASET, THE LUNG DATASET AND THE SRBCT DATASET, RESPECTIVELY.

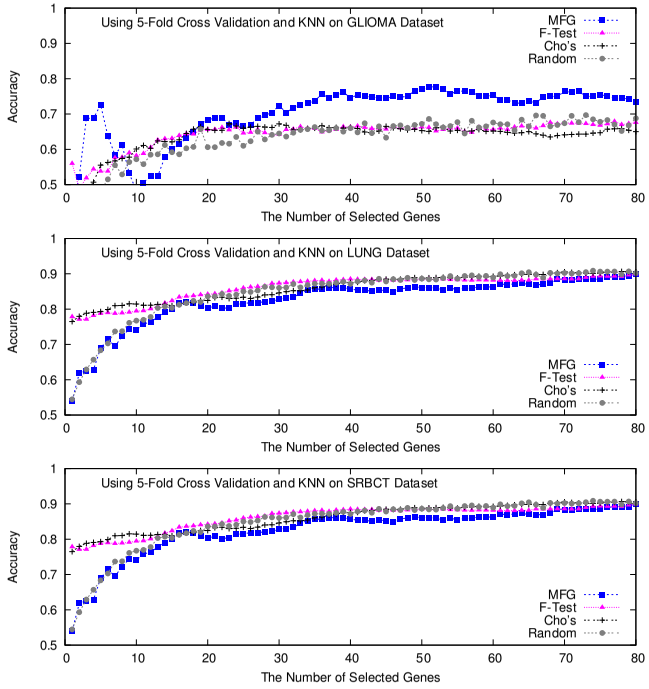

Fig. 4

KNN-CLASSIFIER 5-FOLD CROSS VALIDATION CLASSIFICATION ACCURACIES FOR THE THREE GENE SELECTION METHODS ON THE GLIOMA DATASET, THE LUNG DATASET AND THE SRBCT DATASET, RESPECTIVELY.

IV. DISCUSSION

A. The Curse of Dimensionality

Unlike the general feature selection tasks, gene expression microarray datasets normally involve a huge number of genes but only a small number of samples. Can the MFG method effectively reduce the dimensionality of the gene space? From the SVM-classifier 5-fold cross validation classification accuracies of the MFG method, F-test, Cho's and Random on the three full datasets (Figure 3), we might be able to conclude that 1) when the ratio between the number of genes and the number of samples is large, then the MFG method does perform better in terms of identifying discriminatory genes for class prediction (the GLIOMA dataset); 2) when the ratio is small, then all three gene selection methods perform equally well (the LUNG and the SRBCT datasets). Therefore, we can conclude that in all cases, applying the MFG method to select genes would highly likely result in a subset of good quality discriminatory genes, and these genes can be used to build classifier to give higher class prediction power.

The above conclusions hold for the SVM-classifiers. As we indicated in the above, it seems from Figure 4 that when the KNN classifier do not work particularly well with the MFG method. We conjecture that there could be some correlations between the gene selection method and the classifier, and only when they are "compatible" the resultant class predictor would have a high accuracy. More theoretical analysis should be done to address this issue.

B. The Efficiency

In terms of running time, the MFG method is equally efficient to the simplest gene selection methods such as F-test. It is much faster than SFS and SFFS that intend to find a subset of more discriminatory genes. The experimental results showed that the classification accuracy of the MFG method is consistently higher than those of the SFS and SFFS methods, though only on the reduced datasets. When larger datasets present, the MFG method can successfully return a subset of discriminatory genes while the SFS and SFFS methods would not be able to within a reasonable time frame. In this sense, the MFG method serves as a practical alternative to the SFS and SFFS methods.

C. The Classifier

In the KNN-classifiers we built in the experiments, we set k to 5. We have tested several other values for k and found out that in general $k = 5$ performed the best, in terms of the classification accuracy. Note that there is a majority voting scheme in a KNN-classifier. Consequently, there are only a limited number of choices for k since the minimum numbers of samples in one class in the three datasets are all small. From the above plots, one can easily see that choosing a classifier makes a difference in the final classification accuracy (Figure 3 versus Figure 4). For instance, when combined with the SVM-classifier, the MFG method always has the highest classification accuracy, except when only a small number of

genes were selected, on the LUNG and the SRBCT datasets. When combined with the KNN-classifier, we have seen that the MFG method only performed equally well to the F-test and the Cho's.

V. Conclusions and Future Work

In this paper, we proposed an intuitively simple model-free gene selection method, the MFG method, to implement three heuristics, greedy, exchange and c-kick. The key idea in the MFG method is to use machine learning classifiers to assign each gene a score, namely its classification accuracy, and based on the score, to select genes in an intuitively sound manner. The experimental results on three real microarray datasets showed that, most of the time, the performance of the MFG method was better than F-test and Cho's, and was more stable than SFS and SFFS. We certainly would like to recommend the MFG method as an alternative to existing gene selection methods. Regarding the microarray dataset integration for further analysis, we have started the work to collect a dataset containing more than $1,200$ samples (the same series of Affymetrix chips U95Av2) on about $12,000$ genes, grouped into 14 classes. Further testing the MFG method on this huge dataset is ongoing. We are also looking into possibly excluding some genes that have very close expression profiles to already selected genes from being used in building the classifier. The consideration is that these genes would have very close discrimination powers to the already selected genes and therefore, for the pure classification purpose, using them would not be helpful. Note that this pre-filtering is done before the classification accuracy is calculated and thus could save a big portion of training time.

Lastly, it is recommended that the correlation of expression between class profiles identified by the MFG method should be compared with those by the other gene selection methods, and use it as an alternative comparison measurement.

Acknowledgments

LX's research was done while visiting the University of Alberta and partially supported by AHFMR. YS, ZC, RG and GL are grateful to the research support from AICML, CFI, NSERC and the University of Alberta. All authors would like to thank the three reviewers for their many constructive comments on the submission, most of which have been addressed in this final version.

References

[1] S. Dudoit, J. Fridlyand, and T. P. Speed. Comparison of discrimination methods for the classification of tumors using gene expression data. *Journal of the American Statistical Association*, 97:77–87, 2002.

[2] C. Ding and H. Peng. Minimum redundancy feature selection from microarray gene expression data. In *Proceedings of IEEE Computer Society Bioinformatics Conference (CSB'03)*, pages 523–530, 2003.

[3] J. Cho, D. Lee, J. H. Park, and I. B. Lee. New gene selection for classification of cancer subtype considering within-class variation. *FEBS Letters*, 551:3–7, 2003.

[4] K. Yang, Z. Cai, J. Li, and G.-H. Lin. A stable gene selection in microarray data analysis. *BMC Bioinformatics*, 7:228, 2006.

[5] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17:491–502, 2005.

[6] P. Baldi and A. D. Long. A Bayesian framework for the analysis of microarray expression data: Regularized t-test and statistical inferences of gene changes. *Bioinformatics*, 17:509–519, 2001.

[7] O. G. Troyanskaya, M. E. Garber, P. O. Brown, D. Botstein, and R. B. Altman. Nonparametric methods for identifying differentially expressed genes in microarray data. *Bioinformatics*, 18:1454–1461, 2002.

[8] M. Xiong, X. Fang, and J. Zhao. Biomarker identification by feature wrappers. *Genome Research*, 11:1878–1887, 2001.

[9] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine Learning*, 46:389–422, 2002.

[10] P. Pudil, J. Novovicova, and J. Kittler. Floating search methods for feature selection. *Pattern Recognition Letters*, 15:1119–1125, 1994.

[11] C. L. Nutt, D. R. Mani, R. A. Betensky, P. Tamayo, J. G. Cairncross, C. Ladd, U. Pohl, C. Hartmann, M. E. McLaughlin, T. T. Batchelor, P. M. Black, A. V. Deimling, S. L. Pomeroy, T. R. Golub, and D. N. Louis. Gene expression-based classification of malignant gliomas correlates better with survival than histological classification. *Cancer Research*, 63:1602–1607, 2003.

[12] A. Bhattacharjee, W. G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E. J. Mark, E. S. Lander, W. Wong, B. E. Johnson, T. R. Golub, D. J. Sugarbaker, and M. Meyerso. Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of National Academy of Sciences of the United States of America*, 98:13790–13795, 2001.

[13] J. Khan, J. S. Wei, M. Ringner, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine*, 7:673–679, 2001.