Fast Greedy Searching Informative Genes via Redundancy Bound Bootstrapping

Dong Hua and Abdou Youssef *

Abstract

The identification of informative genes is very important in study of genomics. This task can be interpreted as searching a subset of genes such that an optimal "ratio of quality to price" is achieved. The "quality" refers to the discrimination power of genes and the "price" means the redundancy involved. This problem is NP hard. In contrast to many other methods, we discretize the gene expression profiling in this paper and approximate the optimizing process by combining individual ranking and sequential forward selection together to greedy searching informative genes in the context of a mathematical optimization formularization. The bootstrapping technique is employed to optimize a key parameter involved, namely, the redundancy bound, which reduces the greedy search cost extensively. The performance is evaluated and compared with previous results over publicly available microarray datasets.

Keywords: gene identification, microarray, bootstrapping

1 Introduction

A significant step towards the current information revolution can be appreciated from the successfully applied new techniques and tools in molecular biology and genetics research. Such technologies make it possible to collect biological information rapidly at an unprecedented level of detail in large quantities. Among the most powerful technologies, microarrays [8] provide the tool to extract biological significance such as the changes in expression profiling of genes under distinct types (e.g., normal vs cancer type), which shed the light on use of them in many fields including pharmacogenomics [24], medical diagnostics [25], drug target identification [20] and underlying gene regulatory networks [12]. An important task is to identify informative features (genes) which contribute to the target study (e.g., cancer diagnosis) significantly. This is necessary from the machine learning perspective: when samples are limited and the number of features is very large beyond a certain point, classification accuracy will reduce. Instead of using all features, one may look for a subset, which can most discriminatively and compactly represent the expression patterns. In other words, genes with maximum discrimination power while minimum redundancy are preferred. By doing so, the further cost of study will be reduced and the diagnosis can also be concentrated and improved.

The task of identifying informative genes can also be interpreted as finding a subset of genes which achieve an optimal "ratio of quality to price". Here, the "quality" represents the discrimination power of genes and the "price" means the redundancy involved. The characteristic of microarray, interrogating thousands or tens of thousands of genes simultaneously with limited samples, poses a big challenge in this NP hard [7] problem [1, 2]: searching the absolute optimal subset of genes is impossible in practice. Popular approximation algorithms include individual ranking (IR) [11, 13, 17, 19], that is, rank the genes and choose the top, and sequential forward selection (SFS), i.e., choose the best gene as the seed and add one more per iteration such that the obtained subset maximizes the given criterion function [3, 4, 16, 30]. In contrast to many other papers, we discretize the gene expression profiling and perform the gene selection by combining IR and SFS together with fast bootstrapping for the parameter learning, namely, the redundancy bound, in the context of a mathematical optimization formularization. Specifically, the optimization formularization is interpreted as maximizing the discrimination power of genes ($U(\mathbf{f})$) iteratively with the redundancy bound ($V(\mathbf{f})$) which will be detailed later on.

The rest of this paper is organized as follows. In section 2, we present models and methods. Experiments are performed on real data sets and evaluated in section 3. Section 4 provides the conclusion.

* Address: Department of Computer Science, The George Washington University, 801 22nd St. Suite 704 NW, Washington DC 20052. The first author is also with Bioengineering Department, University of Illinois at Urbana-Champaign. Email: {gwuhua, ayoussef}@gwu.edu

2 Models and Methods

2.1 Mathematical Formularization

Consider a $k(\geq 2)$-class discriminant analysis with p genes, i.e., g_1, g_2, ..., g_p, and n microarray samples involved. Let X_{ij} be the value in terms of the measurement of g_i expression from the jth sample where $i = 1, \ldots, p$ and $j = 1, \ldots, n$. Typically, such microarray data can be written as a form of matrix, $\mathbf{M}$:

$$\mathbf{M} = \begin{pmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ X_{p1} & X_{p2} & \cdots & X_{pn} \end{pmatrix},$$

where the columns and rows correspond to samples and genes, respectively.

Given $\mathbf{M}$, to select m genes out of p genes for discriminant analysis can be viewed as the identification of representative rows (genes) to stand for the entire expression pattern across all the given samples instead of $\mathbf{M}$ itself. Effectiveness can be evaluated from two ways: (1) the combination of chosen rows can differentiate samples distinguishably; and (2) these rows contain redundancy as low as possible. In other words, selected genes should be discriminative and compact simultaneously. Let $\mathbf{f}$ be the selected genes, $U(\mathbf{f})$ the discriminative power of $\mathbf{f}$, and $V(\mathbf{f})$ redundancy in correspond. Generally, larger $U(\mathbf{f})$ implies higher discriminative power; while lower $V(\mathbf{f})$ implies less redundancy. As such, the gene selection is naturally formularized into an optimization problem, which is of the form [1]:

$$maximize: \ U(\mathbf{f}), \ subject\ to: \ V(\mathbf{f}) \leq T(U, V),$$

where T is a threshold function of U and V called "redundancy bound".

2.2 $U(\mathbf{f})$ and $V(\mathbf{f})$ Instantiation

To model $U(\mathbf{f})$, the key is to find some measurement such that the discrimination power of $\mathbf{f}$ is truly expressed. In this paper, we investigate this issue from the statistical point of view. It is assumed that the data $\mathbf{M}$ are normalized so that the genes have mean 0 and variance 1 across samples. Given a fixed gene, let Y_{ij} be the expression level from the j_{th} sample of the i_{th} class. Note that these Y_{ij} come from the corresponding row of $\mathbf{M}$. For example, for g_1, Y_{ij} are a rearrangement of the first row of $\mathbf{M}$. The following general model is considered for Y_{ij} in this paper:

$$Y_{ij} = \mu_i + \epsilon_{ij}, \quad \text{for} \quad i = 1, 2, \ldots, k; \ j = 1, 2, \ldots, n_i$$

where $n_1 + n_2 + \ldots + n_k = n$, μ_i is the mean expression level of the gene in class i, and ϵ_{ij} are the error terms, independent normal random variables with

$$E(\epsilon_{ij}) = 0, \ V(\epsilon_{ij}) = \sigma_i^2 < \infty,$$

$$\text{for } \ i = 1, 2, \ldots, k; \ j = 1, 2, \ldots, n_i.$$

An important task, associated with above model, is to detect whether or not there exists some difference among the means $\mu_1, \mu_2, \ldots, \mu_k$. It is often achieved by certain statistics, the well known ANOVA F test for instance, which is well suited for measuring the discriminative power of genes as thought in this paper. Specifically, given a test statistics $\mathcal{F}$, we define the *discrimination power* of a gene, $d(g_i)$, as the value of $\mathcal{F}$ evaluated over the samples. This definition is based on the fact that with larger $\mathcal{F}$ the null hypothesis $H_0 : \mu_1 = \mu_2 = \ldots = \mu_k$ will be rejected more likely. Therefore, larger $\mathcal{F}$ implies higher discrimination power of the corresponding gene across classes of samples. We also note that discrimination power of genes could be determined equally well via p-values from $\mathcal{F}$. However, due to small sizes n_i, it is hard to justify the approximation of the known distribution to $\mathcal{F}$ and hence p-values may not reflect the real functionality of $\mathcal{F}$. Therefore, the value of $\mathcal{F}$ is preferred.

Usually, if the variances are equal, namely, $\sigma_1^2 = \sigma_2^2 = \ldots = \sigma_k^2$, then it is simply the commonly used one-way ANOVA model and hence the ANOVA F test is the optimal option [18, 22]. For microarray data, the existence of heterogeneity in variances is more realistic, since different σ_i may describe different variation of the gene expression across classes. It makes the above task challenging however, related to the well-known Behrens-Fisher problem [27]. When sample sizes of all classes are equal, i.e. $n_1 = n_2 = \cdots = n_k$, the presence of heterogeneous variances of the errors only slightly affects the F test. If sample sizes are *not* equal, the effect is serious [21]. The actual type I error is inflated when smaller sizes n_i are associated with larger variances σ_i^2. In contrast, the significance levels are smaller than anticipated when larger sizes n_i are associated with larger variances σ_i^2.

In this paper, the parametric Brown-Forsythe test statistic is chosen due to its preferable performance in [6], which is given by [5]:

$$B = \frac{\sum n_i(\bar{Y}_{i\cdot} - \bar{Y}_{\cdot\cdot})^2}{\sum(1 - n_i/n)s_i^2}.$$

[1] Other forms may exist. However, we use this form in this paper for simplicity.

Under H_0, B is distributed approximately as $F_{k-1,\nu}$, where

$$\nu = \frac{[\sum(1-n_i/n)s_i^2]^2}{\sum(1-n_i/n)^2 s_i^4/(n_i-1)}.$$

To model $V(\mathbf{f})$, we simply use Pearson correlation between genes. Given $\mathbf{M}$, the correlation of g_i and $g_{i\prime}$ is given by

$$\rho(g_i,\ g_{i\prime}) = \frac{\sum_j (X_{ij}-\bar{X}_i)(X_{i'j}-\bar{X}_{i'})}{\sqrt{\sum_j (X_{ij}-\bar{X}_i)^2 \sum_j (X_{i'j}-\bar{X}_{i'})^2}}\ ,$$

where $\bar{X}_i = \sum_j X_{ij}/n$ is the average level of g_i, based on the n samples in correspond.

Of particular, we simply use the following optimization formula:

$$maximize:\ U(\mathbf{f}) = \frac{1}{|\mathbf{f}|}\sum_{g_i \in \mathbf{f}} d(g_i)$$

subject to:

$$V(\mathbf{f}) = max\{\rho(g_i,g_j), \forall g_i, g_j \in \mathbf{f}\ and\ i \neq j\} \leq T.$$

T is adjusted dynamically via bootstrapping. This process is called "redundancy bound bootstrapping" in this paper. Algorithm 1 only shows the gene selection given the number of genes needed (m) and a specified T value. The fast optimization of T via bootstrapping is detailed later on (in Subsection 2.5).

Algorithm 1 works as follows. Given a test statistic $\mathcal{F}$, rank all genes with $d(.)$ descending and choose the top as the seed which has the highest discrimination. Consider the rest whose correlation to the chosen gene is below T. Similarly, the top is chosen as the second. And then perform the next iteration. Note that we rank genes only once before the seed selection. As such, the kth informative gene is the one receiving the highest discrimination power from the set of all genes with correlation to each of the chosen $k-1$ genes below T. Above process will be repeated until the given number m of genes are obtained or all the genes have been scanned.

2.3 Classifier

For simplicity, we choose Naive Bayes as the classifier. The purpose of performing classification is to provide the evidence for the quality of selected genes. This may not be necessary in reality. For example, given a requirement of how many genes you need, we may use the algorithm to select genes straightforwardly and then return the index (or names) of the chosen genes.

Algorithm 1 Gene selection algorithm

```
1: function Σ=GeneSel(M, m, T)        ▷ M is the data matrix, Σ is the target feature set
2:     Σ ← φ
3:     M' ← rank(M)                    ▷ Feature sorting
4:     Σ ← Σ ∪ {first feature of M'}   ▷ Choose the first one as the seed
5:     M' ← M'\{first row of M'}       ▷ Remove it
6:     while M' ≠ φ and |Σ| < m do     ▷ Loop for qualified features
7:         if max{ρ(first feature of M', Σ)} ≤ T then   ▷ Check correlation criterion
8:             Σ ← Σ ∪ {first feature of M'}
9:             M' ← M'\{first row of M'}
10:        else
11:            M' ← M'\{first row of M'}
12:        end if
13:    end while
14:    return Σ
15: end function
```

The input for Naive Bayes in this paper is restricted in discrete data. The reason to discretize the data is to remove the noise in some sense. Another reason is that most other papers use continuous data, we use the discrete data for comparison. Experiment results show its effectiveness. The discretization technique is given next.

Consider a k-class ($k \geq 2$) classification issue. Let $\mathbf{X} = (X_1, X_2, \cdots, X_p)^T$ be a p-dimensional feature vector, where T is the transpose operation. We use C to denote the class label of $\mathbf{X}$ with π_i referring to the prior probability, $P(C=i)$, for $i = 1, 2, \cdots, k$. Suppose that given $C=i$, the joint distribution of $\mathbf{X}$ is given by $P_i(\mathbf{X})$. If $\mathbf{x}$ is an observed value of $\mathbf{X}$, then it follows from the Bayes formula that the posterior probability of class i given $\mathbf{X}=\mathbf{x}$ is

$$P(i|\mathbf{x}) = \frac{\pi_i P_i(\mathbf{x})}{\sum_{i=1}^k \pi_i P_i(\mathbf{x})}\ . \tag{1}$$

With $0-1$ loss function, the Bayes rule states that we classify $\mathbf{x}$ to the most probable class according to posterior probabilities, that is, given $\mathbf{X}=\mathbf{x}$, the class label is chosen to be

$$C(\mathbf{x}) = \text{argmax}_{\ i}\ \{P(i|\mathbf{x})\}. \tag{2}$$

The naive Bayes model makes the additional assumption that given a class $C=i$, the features $X_1, X_2, \cdots, X_p$ are independent with each other. Under this assumption, we have, for each i,

$$P_i(\mathbf{x}) = \prod_{l=1}^{p} P_{il}(x_l), \tag{3}$$

where $P_{il}(x_l)$ is the class-conditional density of X_l for $l = 1, 2, \cdots, p$. Using (1), (2) and (3), it is easy to see that the naive Bayes classifier assigns $\mathbf{x}$ into the following class by taking the natural logarithm:

$$C(\mathbf{x}) = \text{argmax}_{\,i} \left\{ \ln \pi_i + \sum_{l=1}^{p} \ln P_{il}(x_l) \right\}. \quad (4)$$

2.4 Discretizer

Equal width interval binning is used. Actually, it is the simplest approach perhaps to achieve the generation of discrete values. Given a continuous value x in an array, the corresponding discrete value is given by:

$$x^d = \lceil \frac{x - x_{min}}{x_{max} - x_{min}} \rceil \times L,$$

where L is customized number of bins. This approach is chosen, besides its simplicity, also because it is a *unsupervised* discretizer which makes no use of any sample class information. Moreover, it often achieves good performance for Naive Bayes classifier vs continuous values as compared by Dougherty et al. (1995) using 16 data sets [10]. Ten-bins are used in this paper.

2.5 Bootstrapping for "T"

Knowing that we have two parameters in Algorithm 1, one is "how many genes" you need (m) and the other is the redundancy bound (T). m is given by the user in this paper, and T depends on the dataset. Optimizing T through the raw data is very expensive given that average $10K$ genes are involved in our datasets. This scale is very common in microarray experiments. To estimate the T fast, we employ the bootstrapping technique to achieve "redundancy bound bootstrapping". Its effectiveness is shown in Section 3. Algorithm 2 shows the detailed procedure.

It works as follows: given a raw data matrix $\mathbf{M}$, we randomly select a number of rows (m_{bst}) and all sample columns from $\mathbf{M}$ to form a new data matrix $\mathbf{M}'$. $\mathbf{M}'$ is usually much smaller than $\mathbf{M}$ given a small value m_{bst} (e.g., 100). We greedy search the optimal redundancy bound over $\mathbf{M}'$ by examining the cross-validation table in an exhausting manner. The cross-validation table consists of a number of entries which give the leave-one-out cross-validation (LOOCV) errors for any instantiation of m and T (e.g., $m = 20$ and $T = .5$ shown in Figure 1). In our experiments, m is chosen from 1 to 50 and T chosen from .05 to 1 with step length .05. We locate the region within which the average error is minimal. This region is represented by a rectangular box consisting of consecutive T

Algorithm 2 Bootstrapping algorithm for learning redundancy bound T

function T=**TBstrap**($\mathbf{M}, m_{bst}, cnt_{bst}, m_{max}$) ▷ $\mathbf{M}$ is the data matrix, m_{bst} is the number of genes chosen for bootstrapping, cnt_{bst} is the times for bootstrapping and m_{max} the maximum number of genes chosen for the target feature (gene) set
  $V_{arr} \leftarrow 0$ ▷ The l_{th} entry is the voting value for the l_{th} T value
  for $i = 1$ *to* cnt_{bst} **do**
    choose m_{bst} rows from $\mathbf{M}$ randomly and form a new data matrix $\mathbf{M}'$
    $Ta_{CV} \leftarrow \phi$ ▷ A table storing the LOOCV error for each instantiation of m and T
    for $j = 1$ *to* m_{max} **do**
      for $k = .05$ *to* 1 with step $.05$ **do**
        Σ=GeneSel($\mathbf{M}, j, k$)
        $err \leftarrow LOOCVerror$
        $Ta_{CV}[i, j] \leftarrow err$
      end for
    end for
    Locate the region (5 consecutive T values vs 10 consecutive m values) achieving minimum average LOOCV error
    for each entry in V_{arr} **do**
      if the l_{th} T value belongs to the region **then**
        $V_{arr}[l] \leftarrow V_{arr}[l] + 1$
      end if
    end for
  end for
  return argmax($V_{arr}[l]$)*.05 ▷ .05 is the step length for T increment which can be specified by users
end function

values and m values. The number of T values or m values can be specified by users. Here, we use 5×10, i.e., 5 consecutive T values and 10 consecutive m values. If there are more than one region achieving the minimal average LOOCV errors, we choose that with smallest starting values of T and m intervals. We perform above process cnt_{bst} (10 by default) times and take voting to get the optimal T value for the raw data. The entire procedure is very fast given that the size of data matrix $\mathbf{M}'$ during bootstrapping is very small.

3 Experimental Results

We perform the experiments on several publicly available datasets. To show the quality of selected genes, the leave-one-out cross-validation (LOOCV) is used. Smaller LOOCV error means better quality. The maximum number of genes (m) is allowed to be 50 in our experiments. The number of genes (m_{bst}) for bootstrapping is 100 and the times of bootstrapping (cnt_{bst}) is 10. Datasets include Ovarian [28], MLL Leukemia [26], Lung Cancer [14], and Colon [2]. The information about the number of genes and samples are shown in Table 1.

We evaluate our algorithms from two perspectives:

1. Does the gene selection algorithm still achieves superior performance with redundancy bound bootstrapping by comparing with the previous best result?

2. Does the gene selection algorithm improves the search speed significantly by incorporating redundancy bound bootstrapping as compared with the previous greedy algorithm [15] which does not use the bootstrapping technique?

3.1 Ovarian

This dataset contains 36 samples including 5 normal tissues, 27 epithelial ovarian tumor samples, and 4 malignant epithelial ovarian cell lines. 7129 genes are employed. By the bootstrapping, we obtain the optimal redundancy bound $.35$. 0 LOOCV error is obtained. For showing the effectiveness of the bootstrapping technique for T value estimation, we summarize all LOOCV errors over the raw data using the greedy algorithm [15] in Figure 1 which is not necessary in reality. The numbers of the first line (from .05 to 1) correspond to different redundancy bound T, and the number (m) of the first column (e.g., 45) correspond to the number of genes required. The bottom line, i.e., $\hat{m}$ corresponds to the real maximum number of genes that can be chosen under the redundancy bound T. The rest entries are LOOCV errors. Smaller values indicate higher quality of genes selected. The rectangular boxes are raised by the bootstrapping in Algorithm 2). It is easy to find that the region overlapped extensively by these boxes across the T value are from .3 to .4. This interval bounds the optimal T value ($.35$). In terms of search speed, if we do not employ the bootstrapping technique (Algorithm 2), we have to examine all possible T values (.05 to 1 with step length .05) to find the optimal T value over the huge raw data like the previous greedy search algorithm [15]. The time needed is about 55 min. In contrast, only 3 min is needed with using bootstrapping technique in this paper.

3.2 MLL Leukemia

This dataset contains both training data and test data. In this paper, we combine them together for LOOCV. Training data summarizes 57 leukemia samples (20 ALL, 17 MLL and 20 AML). Testing data summarizes 4 ALL, 3 MLL and 8 AML samples. Hence there are 3 classes in total. The number of genes is 12582.

In the same way, the LOOCV result over the raw data by the greedy algorithm [15] is summarized in Figure 2. The optimal T value by bootstrapping is .5. 100% accuracy, i.e., 0 errors in cross-validation, is also achieved. Similar as before, the rectangular boxes shown in the figure are raised by the bootstrapping Algorithm 2. The time with or without bootstrapping technique, namely, the previous greedy search algorithm [15], are 7.8 min vs 149.8 min.

3.3 Lung Cancer

For this data, the classification is performed between two classes, i.e., malignant pleural mesothelioma (MPM) and adenocarcinoma (ADCA) of the lung. 32 training samples and 149 testing samples are combined together. As such, there are 31 MPM and 150 ADCA described by 12533 genes. The best result (100% accuracy) is achieved under $T = .25$ found via the bootstrapping. The time with or without bootstrapping technique are 16.3 min vs 327.6 min.

3.4 Colon

There are 62 samples in this dataset collected from colon-cancer patients. Among them, 40 tumor biopsies are from tumors, labelled with 'negative' and 22 normal biopsies, labelled with 'positive', from healthy parts of the colons of the same patients. Nearly 7000 genes are involved. 2 LOOCV errors ($T = .45$) is achieved. It is better than the best result achieved previously (4 LOOCV errors). The time with or without bootstrapping technique are 2.6 min vs 49.7 min.

m	T=.05	.1	.15	.2	.25	.3	.35	.4	.45	.5	.55	.6	.65	.7	.75	.8	.85	.9	.95	1
1	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
2	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	4	4	4
3	2	2	2	2	2	2	2	2	2	2	2	2	2	5	5	4	4	5	4	4
4	1	2	1	2	0	1	1	1	1	1	2	2	2	1	0	5	5	1	4	4
5	2	2	2	2	0	0	0	1	1	1	0	0	2	0	0	1	1	1	2	2
6	4	2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	4	1	2	2
7	3	3	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	2	2
8	4	2	0	2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
9	4	2	1	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
10	4	2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
11	4	4	2	1	1	0	1	0	1	0	0	0	0	0	0	0	0	0	1	1
12	4	2	2	4	1	0	1	0	0	1	0	0	0	0	0	0	0	0	1	1
13	4	2	2	4	1	0	2	0	0	0	0	0	0	0	0	0	0	0	0	1
14	4	2	3	3	1	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0
15	4	2	2	4	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
16	4	2	4	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	4	2	3	6	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	4	2	4	2	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	4	2	4	3	2	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
20	4	2	4	4	3	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
21	4	2	4	4	3	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
22	4	2	4	3	5	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
23	4	2	4	3	4	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
24	4	2	4	3	4	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
25	4	2	4	3	4	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0
26	4	2	4	6	5	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0
27	4	2	4	5	6	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
28	4	2	4	5	3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
29	4	2	4	6	5	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
30	4	2	4	6	5	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
31	4	2	4	7	6	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
32	4	2	4	8	4	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
33	4	2	4	8	4	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
34	4	2	4	8	4	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
35	4	2	4	8	4	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0
36	4	2	4	8	3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
37	4	2	4	8	3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
38	4	2	4	8	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
39	4	2	4	8	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
40	4	2	4	8	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
41	4	2	4	8	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
42	4	2	4	8	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
43	4	2	4	8	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
44	4	2	4	8	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
45	4	2	4	8	5	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
46	4	2	4	8	5	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0
47	4	2	4	8	5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
48	4	2	4	8	5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
49	4	2	4	8	5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
50	4	2	4	8	5	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
$\hat{m}$	9	12	18	32	47	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50

Figure 1: Ovarian: Experiment result in terms of accuracy with LOOCV over the raw data. The numbers of the first line (from .05 to 1) correspond to different redundancy bound T, and the number of the first column (e.g., 45) correspond to different the number of genes need to be chosen. $\hat{m}$ refers to the maximum number of genes that could be chosen under the corresponding T value. The rectangular boxes are raised via bootstrapping in Algorithm 2.

m	T=.05	.1	.15	.2	.25	.3	.35	.4	.45	.5	.55	.6	.65	.7	.75	.8	.85	.9	.95	1
1	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17	17
2	17	17	17	17	17	17	17	17	17	17	17	13	13	13	13	16	16	16	16	16
3	12	14	14	14	14	7	7	15	14	14	15	10	10	10	12	12	9	9	9	9
4	10	10	10	13	9	6	8	6	7	14	14	11	11	11	11	10	7	7	7	7
5	10	11	11	11	6	6	5	7	4	4	10	8	10	8	10	7	6	6	6	6
6	9	9	12	9	6	4	5	5	5	2	9	6	6	10	7	8	7	7	7	7
7	12	10	11	5	8	5	5	3	5	1	8	4	7	12	10	7	6	6	6	6
8	13	12	7	6	6	5	4	4	4	3	8	4	7	11	8	5	8	7	7	7
9	13	13	8	7	7	3	3	4	4	1	9	3	5	8	6	4	9	8	8	8
10	13	8	9	4	6	4	2	4	4	2	8	3	8	6	8	4	8	7	7	7
11	14	7	10	5	6	3	4	3	4	1	8	3	6	7	8	4	8	8	8	8
12	14	14	9	7	8	3	4	1	2	0	8	1	6	7	8	4	6	10	10	10
13	14	12	15	5	9	2	3	3	3	0	6	1	6	8	8	5	4	7	7	7
14	14	13	13	5	10	2	3	3	3	0	5	1	5	8	8	6	3	6	6	6
15	14	12	12	7	7	2	5	3	2	0	4	1	6	7	9	6	4	4	4	4
16	14	11	12	9	9	3	5	4	2	0	7	1	6	8	8	4	5	4	4	4
17	14	13	10	9	8	2	4	4	2	1	6	1	6	6	8	6	5	4	4	4
18	14	11	11	9	7	2	4	2	2	1	4	0	6	4	7	5	5	4	4	4
19	14	11	11	8	7	2	4	4	2	1	5	0	6	6	6	5	4	4	4	4
20	14	10	12	7	8	1	3	3	3	0	5	0	6	5	7	5	4	4	4	4
21	14	11	11	7	9	1	3	3	3	0	5	0	7	5	6	5	5	4	4	4
22	14	11	11	7	6	2	2	3	1	0	4	0	4	2	8	4	5	4	4	4
23	14	11	12	9	5	2	2	3	3	0	4	1	3	2	7	5	5	4	4	4
24	14	11	7	9	6	2	1	3	3	1	3	1	1	2	6	5	5	5	5	5
25	14	11	10	8	6	0	2	3	1	0	3	1	0	2	6	4	5	5	5	5
26	14	11	10	9	6	0	3	3	2	0	3	1	0	2	5	4	5	5	5	5
27	14	11	10	7	5	1	2	3	3	0	3	1	0	2	6	4	5	6	6	6
28	14	11	10	9	8	1	3	3	2	0	3	1	0	2	7	5	5	6	6	6
29	14	11	11	6	9	2	3	3	2	0	3	1	0	2	4	5	4	6	6	6
30	14	11	14	5	8	2	3	3	2	1	3	2	0	1	3	4	4	6	6	6
31	14	11	14	6	8	1	3	3	2	1	3	2	0	1	4	4	4	6	6	6
32	14	11	14	7	8	3	3	3	3	1	3	3	1	1	3	4	4	6	6	6
33	14	11	14	8	7	3	3	3	2	1	3	2	1	1	3	3	4	5	5	5
34	14	11	14	7	7	2	3	3	2	0	3	2	2	1	3	2	3	6	6	6
35	14	11	14	6	7	1	3	3	2	0	3	2	0	2	3	4	3	5	5	5
36	14	11	14	6	7	2	3	3	2	0	3	2	0	2	3	4	4	5	5	5
37	14	11	14	6	7	2	3	3	2	0	3	0	0	1	3	4	4	4	4	4
38	14	11	14	6	7	2	3	2	1	0	2	0	1	1	3	4	4	6	6	6
39	14	11	14	6	7	2	3	2	2	0	2	0	1	2	3	4	4	5	5	5
40	14	11	14	6	7	2	3	1	2	0	3	0	1	1	3	4	4	4	4	4
41	14	11	14	8	7	2	3	2	2	0	2	1	1	1	3	3	3	6	6	6
42	14	11	14	8	7	2	3	2	3	0	2	0	1	1	3	3	3	5	5	5
43	14	11	14	8	6	2	3	4	3	0	1	1	1	1	3	2	3	5	5	5
44	14	11	14	8	6	2	3	4	3	0	2	0	1	1	3	2	3	5	5	5
45	14	11	14	7	7	2	4	4	2	0	2	1	1	1	3	3	3	5	5	5
46	14	11	14	8	6	2	4	3	2	0	2	1	1	1	3	3	3	5	5	5
47	14	11	14	8	7	2	4	3	2	0	2	1	2	1	3	3	4	5	5	5
48	14	11	14	8	6	3	3	3	2	0	2	1	0	0	3	2	4	5	5	5
49	14	11	14	10	6	3	3	3	2	0	2	1	1	0	3	2	4	5	5	5
50	14	11	14	10	6	2	2	5	1	0	2	1	2	2	2	2	4	5	5	5
$\hat{m}$	11	21	31	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50

Figure 2: MLL Leukemia: Experiment result in terms of accuracy with LOOCV over the raw data. The numbers of the first line (from .05 to 1) correspond to different redundancy bound T, and the number of the first column (e.g., 45) correspond to different the number of genes need to be chosen. $\hat{m}$ refers to the maximum number of genes that could be chosen under the corresponding T value. The optimal T value (.5) is found via bootstrapping in Algorithm 2 which raised the rectangular boxes.

Table 1: Gene expression datasets

Dataset	Ovarian	MLL Leukemia	Lung Cancer	Colon
No of genes	7129	12582	12533	7000
No of samples	68	72	181	62

Table 2: Comparison of the performance (LOOCV errors)

Dataset	Ovarian	MLL Leukemia	Lung Cancer	Colon
Previous Best Result	0	0	2	4
Our Best Result	0	0	0	2

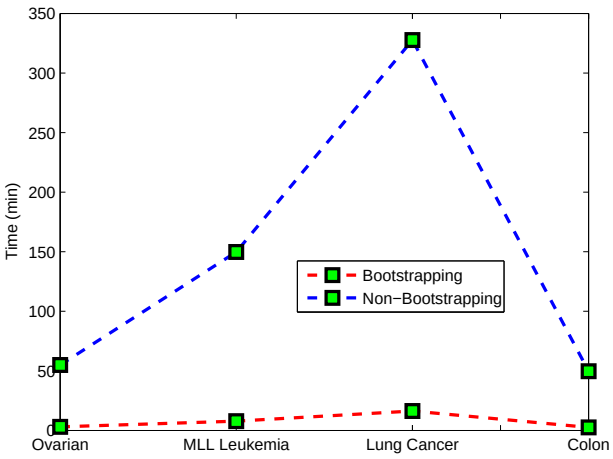

Figure 3: Comparison of search speed with and without bootstrapping algorithm (Algorithm 2).

Comparing with the previous greedy algorithm [15], best performances are still achieved by incorporating redundancy bound bootstrapping. As shown in Table 2, the LOOCV performances are same as before for Ovarian and MLL Leukemia datasets, and better than before for Lung cancer and Colon datasets.

In terms of search speed, Figure 3 compares the greedy search time with and without bootstrapping technique. As mentioned before, if we do not use the bootstrapping technique, namely, the previous greedy algorithm [15], we have to examine the instantiation of m (1 to 50) and T (.05 to 1 with step length .05) exhaustedly to identify the optimal redundancy bound. In contrast, by using Algorithm 2, we can locate the optimal redundancy bound fast, with which we only need to examine one column of the cross-validation table shown in Figure 1. It is clear that the algorithm using bootstrapping technique is much faster.

4 Conclusion

Gene selection is essentially an NP-complete problem. It is important to identify informative genes either in study or diagnosis. In this paper, we formularize this optimization issue by maximizing the discrimination power of genes in terms of Brown-Forsythe statistic, with the redundancy bound modeled by Pearson correlation. The bootstrapping technique is employed for optimizing the redundancy bound in a fast way. Experiments show its effectiveness not only in best accuracy achieved as before on Ovarian and MLL Leukemia datasets, or better than before on Lung Cancer and Colon datasets but also in significantly improved search speed as compared with previous greedy algorithm [15] with no bootstrapping technique. In our future work, we will explore the possibility of bootstrapping technique to optimize the number of genes (m) which is given by manual (users) in this paper.

References

[1] Alizadeh, A., et. al. (2000) Identification of clinically distinct types of diffuse large B-cell lymphoma based on gene expression patterns. *Nature*, 403, 503-511.

[2] Alon, U., Barkai, N., Notterman, D., Gish, K., Ybarra, S., Mack, D., and Levine, A. (1999) Broad patterns of gene expression revealed by clustering of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA*, 96, 6745-6750.

[3] Ambroise, C. and McLachlan, G. (2002) Selection bias in gene extraction on the basis of microarray gene-expression data. *Proc. Natl. Acad. Sci. USA*, vol. 99 (10), 6562-6566.

[4] Bo, T. and Jonassan, I. (2002) New features subset selection procedures for classification of expression profiles. *Genome Biology*, 3, 0017.1-0017.11.

[5] Brown, M. B., and Forsythe, A. B. (1974) The small sample behavior of some statistics which test the equality of means. *Technometrics*, 16, 129-132.

[6] Chen, D., Liu, Z., Ma, X., and Hua, D.(2005). Selecting genes by test statistics. *Journal of Biomedicine and Biotechnology*, 2, pp. 132–138, June.

[7] Cover, T. and Campenhout, J. (1977) On the possible orderings in the measurement selection problem. *IEEE Trans. Systems, Man, and Cybernetics*, SMC-7(9), 657-661.

[8] Der, S. D., Zhou, A., Williams, B. R. G., and Silverman, R. H. (1998) Identification of genes differentially regulated by interferon alpha, beta, or gamma using oligonucleotide arrays.*Proc. Natl. Acad. Sci. USA*, 95, 15623-15628.

[9] Ding, C. and Peng, H. (2003) Minimum Redundancy Feature Selection from Microarray Gene Expression Data. *CSB 2003*, 523-529.

[10] Dougherty, J., Kohavi, R., and Sahami, M. (1995) Supervised and unsupervised discretization of continuous features, In *Proc. 12th Int. Conf. on Machine Learning (ICML-95)* , 194-202.

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

[12] Eisen, M. and Brown, P. (1999) DNA arrays for analysis of gene expression. *Methods in Enzymology*, 303,179-205.

[13] Furey, T., Cristianini, N., Duffy, N., Bednarski, D., Schummer, M., and Haussler, D. (2000) Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16, 906-914.

[14] Gavin J. Gordon, et. al. (2002) Translation of Microarray Data into Clinically Relevant Cancer Diagnostic Tests Using Gege Expression Ratios in Lung Cancer And Mesothelioma. *Cancer Research*, 62:4963-4967

[15] Hua, D., Chen, D. and Youssef, A. (2005) Identifying genes with the concept of customization. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05), Workshop on Bioinformatics*, Vol 03, pp. 144.

[16] Inza, I., Sierra, B., Blanco, R., and Lerranaga, P. (2002) Gene selection by sequential search wrapper approaches in microarray cancer class prediction, *Journal of Intelligent and Fuzzy Systems*, in press.

[17] Iyer, V., Eisen, M., Ross, D., et al. (1999) The transcriptional program in the response of human fibroblasts to serum, *Science*, vol. 283, 83-87.

[18] Lehman, E. L. (1986). *Testing Statistical Hypotheses*, 2nd edition. New York: Wiley.

[19] Long, A., Mangalam, H., Chan, B., Tolleri, L., Hatfield, G., and Baldi, P. (2001) Improved statistical inference from DNA icroarray data using analysis of variance and a Bayesian statistical framework, *J. Biol. Chem.*, 276, 19937-19944.

[20] Marton, M. J., et. al. (1998). Drug target validation and identification of secondary drug effects using DNA microarrays. *Nature Medicine*, 4(11), 1293-1301.

[21] Montgomery, D. C. (2001). *Design and Analysis of Experiments*, 5th edition. New York: Wiley.

[22] Neter, J., Kutner, M. H., Nachtsheim, C. J., and Wasserman, W. (1996). *Applied Linear Statistical Models*, 4th edition. McGraw-Hill.

[23] Ripley, B. D. (1996) *Pattern Recognition and Neural Networks*, Cambridge University Press.

[24] Roses, A.D.(2000). Pharmacogenetics and the practice of medicine. *Nature,* 405, 857-865.

[25] Ross, D. T., et. al. (2000). Systematic Variation in Gene Expression Patterns in Human Cancer Cell Lines. *Nature Genetics*, 24, 227-235.

[26] Scott A. Armstrong1, et. al. (2002) MLL Translocations Specify A Distinct Gene Expression Profile that Distinguishes A Unique Leukemia, *Nature Genetics*, 30:41-47.

[27] Stuart, A., Ord, J. K., and Arnold, S. (1999). *Advanced Theory of Statistics, Volume 2A: Classical Inference and the Linear Model,* 6th edition. London: Oxford University Press.

[28] Welsh, J. B., Zarrinkar, P. P., Sapinoso, L. M., Kern, S. G., Behling, C. A., Burger, R. A., Monk, B. J., and Hampton, G. M. (2001). Analysis of gene expression in normal and neoplastic ovarian tissue samples identifies candidate molecular markers of epithelial ovarian cancer. *Proc. Natl. Acad. Sci. U.S.A.*, 98:1176-1181.

[29] Xing, E., Jordan, M., and Karp, R. (2001) Feature selection for high-dimensional genomic microarray data, In *Proceedings of Eighteenth International Conference on Machine Learning*, San Francisco.

[30] Xiong, M., Li, W., Zhao, J., Jin, L., and Boerwinkle, E. (2001) Feature (gene) selection in gene expression-based tumor classification, *Mol Genet Metab* vol. 73, 239-247.