Improved shrunken centroid method for better variable selection in cancer classification with high throughput molecular data

by

Xukun Li

B.S., Xiamen University, China, 2014

A REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2017

Approved by:

Major Professor
Dr. Haiyan Wang

Abstract

Cancer type classification with high throughput molecular data has received much attention. Many methods have been published in this area. One of them is called PAM (nearest centroid shrunken algorithm), which is simple and efficient. It can give very good prediction accuracy. A problem with PAM is that this method selects too many genes, some of which may have no influence on cancer type. A reason for this phenomenon is that PAM assumes that all genes have identical distribution and give a common threshold parameter for genes selection. This may not hold in reality since expressions from different genes could have very different distributions due to complicated biological process. We propose a new method aimed to improve the ability of PAM to select informative genes. Keeping informative genes while reducing false positive variables can lead to more accurate classification result and help to pinpoint target genes for further studies. To achieve this goal, we introduce variable specific test based on Edgeworth expansion to select informative genes. We apply this test on each gene and select some genes based on the result of the test so that a large number of genes will be excluded. Afterward, soft thresholding with cross-validation can be further applied to decide a common threshold value. Simulation and real application show that our method can reduce the irrelevant information and select the informative genes more precisely. The simulation results give us more insight about where the newly proposed procedure could improve the accuracy, especially when the data set is skewed or unbalanced. The method can be applied to broad molecular data, including, for example, lipidomic data from mass spectrum, copy number data from genomics, eQLT analysis with GWAS data, etc. We expect the proposed method will help life scientists to accelerate discoveries with highthroughput data.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my sincere gratitude to the Kansas State University and the Statistics Department for the opportunity to learn a fascinating subject. In particular I would like to thank my advisor Dr. Wang for her support, encouragement, and guidance during my research. I also thank Dr. Neill and Dr. Wu for their role as my committee members, as well as for teaching some of my favorite classes. Especially, I would also like to give many thanks to Huaiyu Zhang, who gave valuable suggestions to this research.

Chapter 1

Introduction

It is important to determine the type of cancer accurately. However, the curse of dimensionality, the small number of samples, and the number of irrelevant genes of high throughput genomics data make this task difficult. Beyond prediction accuracy, another important task is to select relevant genes. One straightforward approach is to apply a standard t-test [1;2] to each gene or a non-parametric test such as the Wilcoxon score test [1;3]. Some other machine learning algorithms, such as Support Vector Machine (SVM) [4], k-Top Scoring Pair (k-TSP) [5], have been applied to this area. In this report, we focus on the popular method nearest shrunken centroid [6].

The nearest shrunken centroid method is a method modified from nearest-centroid classification to suit high dimensional data. It is also referred as PAM (prediction analysis of micro-array). PAM measures the scaled distance between each class center and overall center. This distance is thresholded by soft thresholding procedure. The soft-thresholding for distance d is defined as:

$$d' = sign(d)(d-c)I_{(d>c)}, \quad \text{where} \quad I_{(d>c)} = \begin{cases} 1 & \text{if } d > c \\ 0 & \text{otherwise} \end{cases}$$

Where c is the threshold value. After soft thresholding, some distances become zero which means the corresponding variables will be left out in further classification. The threshold value in PAM is chosen by 10-fold cross-validation (CV).

The PAM is a simple and efficient method that has very good prediction accuracy. The paper introducing this method has been cited 2249 times by the time that we worked on this paper. An example of application of PAM is on classification and prediction of clinical Alzheimers diagnosis[7] and for predicting patient Survival[8]. When it is applied to cancer studies, a potential problem is that there may be too many genes being selected by PAM, some of which may not contribute to the cancer type prediction. In the cross-validation step in PAM, a common threshold value is selected for all the genes, which implicitly assumes that all genes have identical distribution. However, this assumption may not hold because the expressions of genes may have very different distributions due to complicated biological processes. To resolve the problem, we propose an individual thresholding procedure which is able to account for different distributional information of different variables. The procedure aims to find informative genes and reducing the number of false positive variables, which can lead to more accurate classification result and help pinpoint target genes for further studies.

We propose to achieve the threshold of variables with individual threshold parameter via a test based on Cornish-Fisher expansion (TCF)[9]. Application of the TCF test to data from each variable will give us a unique critical value. This value is then used along with the soft thresholding procedure to define a unique optimal thresholding parameter for each variable.

Chapter 2

Methods

This chapter describes the details of the new method. For convenience, we refer the new method as TCF-PAM. The overall idea of TCF-PAM is to first apply the test based on Cornish-Fisher expansion to the data from each variable to obtain a critical value. The critical value and any given threshold value will jointly define a thresholding parameter. Why different thresholding parameters are needed? We explain the rationale through the PAM procedure.

In PAM, the class label is decided by the probability given by Naive Bayes algorithm

$$P(class\ k|\mathbf{X} = \mathbf{x}) = \frac{\prod_{i=1}^{n} f_{X_i}(x_i|class\ k)P(class\ k)}{P(\mathbf{X} = \mathbf{x})},$$

where $\mathbf{X}$ is the gene expression data and class k is the class label of the sample. But it is not easy to decide the distribution of $f_{X_i}(x_i|class\ k)$. Suppose there are only 2 classes in the dataset. When both classes follows normal distribution, the

$$log f_{X_i}(x_i|class\ k) \propto \frac{x_i\mu_i - \mu_i^2/2}{\sigma^2} + C_k(x_i, \sigma_i^2),$$

Where $C_k(x_i, \sigma_i^2) = \frac{x_i^2}{2\sigma_i^2} - log(2\pi\sigma_i^2)/2$. In the original PAM algorithm, the term $C_k(x_i, \sigma^2)$ is ignored in further calculation of the discriminant function, which separates the class boundaries. This makes sense in the case with normal data with constant variance. When assuming common σ^2, then the term $C_k(x_i, \sigma^2)$ gives the same value for $k = 1,\ 2$. In this case, there is no point to include $C_k(x_i, \sigma^2)$ in comparison of the likelihood to form discriminant functions. When the data for both classes follows the same gamma distribution,

$$logf_{X_i}(x_i|class\ k) \propto \frac{x_i/\mu_i + log(1/\mu_i)}{1/\nu} + C_k(x_i, \nu),$$

Where $C_k(x, \nu) = \nu log(x\nu) - log(\pi\nu) - log(x)$. In this situation, the $C_k(x, \nu)$ does not play a role. In the two situations ablove, common threshold value for both class are reasonable. When the data for one class is from normal distribution, and the other is from gamma distribution, then

$$logf_{X_i}(x_i|class\ 1) \propto \frac{x_i\mu_i - \mu_i^2/2}{\sigma^2} + C_1(x_i, \sigma^2),$$

$$logf_{X_i}(x_i|class\ 2) \propto \frac{x_i/\mu_i + log(1/\mu_i)}{1/\nu} + C_2(x_i, \nu),$$

where $C_1(x, \sigma^2) = \frac{x^2}{2\sigma^2} - log(2\pi\sigma^2)/2$ and $C_2(x, \nu) = \nu log(x\nu) - log(\pi\nu) - log(x)$. They are different and both contribute to the difference in the likelihood functions. In this case, ignoring the $C_k(x_i, .)$ term in PAM would miss some important information due to the difference in distribution. Unfortunately, we do not know the distribution of data in practice. We do know that the data from genomics are often skewed since they are generally measuring light intensities from high throughput equipment.

To describe how we can improve PAM, let x_{ij} be the gene expression data for i^{th} gene of j^{th} sample, and y_j be the label for the j^{th} sample, where $i = 1, 2, ..., m,\ j = 1, 2, ..., n$. The possible outcomes of y are $1, 2, ..., K$, where K denotes the number of classes in this data set. Let n_k denote the number of samples in class k. The center of i^{th} gene in class k is $\overline{x}_{ik} = \sum_{l \in \{j:\ y_j = k\}} x_{il}/n_k$, and the overall center of i^{th} gene is $\overline{x}_i = \sum_{j=1}^{n} x_{ij}/n$, where $n = \sum_{k=1}^{K} n_k$.

To scale the class center toward the overall center, define

$$d_{ik} = \frac{\overline{x}_{ik} - \overline{x}_i}{m_k\,(s_i + s_0)}$$

where s_i^2 is the mean squared error for the i^{th} gene:

$$s_i^2 = \frac{1}{n-k} \sum_{k=1}^{K} \sum_{l \in \{j:\ y_j = k\}} (x_{il} - \overline{x}_{ik})^2,$$

and $m_k = \sqrt{1/n_k + 1/n}$. In PAM, s_0 is the median of s_i to avoid the situation where the denominator of d_{ik} is equal to 0. We set $s_0 = 10^{-8}$ which is a small value so that the statistic d_{ik} will follow its original distribution.

We modify the PAM method by introducing a selection procedure based on Edgeworth expansion of two-sample t-statistic. Note that the numerator of d_{ik} can be written as:

$$\begin{aligned}
\overline{x}_{ik} - \overline{x}_i &= \frac{\sum_{l \in \{j:\ y_j = k\}} x_{il}}{n_k} - \left(\frac{\sum_{l \in \{j:\ y_j = k\}} x_{il}}{n} + \frac{\sum_{l \in \{j:\ y_j \neq k\}} x_{il}}{n} \right) \\
&= (\frac{1}{n_k} - \frac{1}{n}) n_k \frac{\sum_{l \in \{j:\ y_j = k\}} x_{il}}{n_k} - \frac{n - n_k}{n} \frac{\sum_{l \in \{j:\ y_j \neq k\}} x_{il}}{(n - n_k)} \\
&= \frac{\sum_{l \in \{j:\ y_j = k\}} (1 - \frac{n_k}{n}) x_{il}}{n_k} - \frac{\sum_{l \in \{j:\ y_j \neq k\}} (1 - \frac{n_k}{n}) x_{il}}{n - n_k} \\
&= \overline{z}_{1ik} - \overline{z}_{2ik},
\end{aligned}$$

where $\overline{z}_{1ik}$ is the sample mean of $z_{1ikl} = (1 - n_k/n)\, x_{il}$, for $l \in \{j : y_j = k\}$, and $\overline{z}_{2ik}$ is the sample mean of $z_{2ikl} = (1 - n_k/n)\, x_{il}$, for $l \in \{j : y_j \neq k\}$. The denominator of d_{ik} is basically the estimated standard error of the numerator. From the above derivation, d_{ik} is studentized two-sample statistic for testing the equality of population means for $\{z_{1ikl}, l \in (j : y_j = k)\}$ and $\{z_{2ikl}, l \in (j : y_j = k)\}$. The t-statistic d_{ik} admits the Edgeworth expansion for the cumulative distribution[10;11].

$$P(T < x) = \Phi\,(x) + \frac{1}{\sqrt{N}} \frac{A}{6} (2x^2 + 1) \phi(x) + O(N^{-1})$$

where T is the two sample statistic of $\{z_{1ikl}, l \in (j : y_j = k)\}$ and $\{z_{2ikl}, l \in (j : y_j = k)\}$, $\Phi(x)$ and $\phi(x)$ are CDF and PDF of standard normal distribution and

$$A = \left\{ \frac{\sigma_1^2}{\lambda} + \frac{\sigma_2^2}{1-\lambda} \right\}^{-3/2} \left\{ \frac{\sigma_1^3 \gamma_1}{\lambda^2} - \frac{\sigma_2^3 \gamma_2}{(1-\lambda)^2} \right\},$$

where $\lambda = \dfrac{n_1}{n}$, σ_1^2 and σ_2^2 are the population variance of $\{z_{1ikl}, l \in (j : y_j = k)\}$ and $\{z_{2ikl}, l \in (j : y_j = k)\}$, respectively, and γ_1 and γ_2 are population skewness from the two samples with observed data $\{z_{1ikl}, l \in (j : y_j = k)\}$ and $\{z_{2ikl}, l \in (j : y_j = k)\}$, respectively. Given the Edgeworth expansion, we can find the percentile $\eta_{ik,\ \alpha}$ of the approximated distribution of d_{ik}, which has the following Cornish-Fisher Expansion:

$$\eta_{ik,\ \alpha} = z_\alpha - \frac{1}{\sqrt{N}} \frac{A}{6} (2z_\alpha^2 + 1),$$

where z_α is the α percentile of the standard normal distribution. Since all the population parameters are unknown, the coefficients in the Cornish-Fisher expansion are estimated based on sample moments of z_{1ik} and z_{2ik}. The population variance in A can be estimated by sample variance and population skewness γ_j in A can be estimated by:

$$\hat{\gamma}_j = \frac{n_j}{(n_j-1)(n_j-2)} \sum_{l=1}^{n_j} \left\{ \frac{z_{ikjl} - \bar{z}_{ikj}}{S_j} \right\}^3.$$

It has been shown in Zhang & Wang[12] that the two-sample test based on Cornish-Fisher expansion is more power than the two-sample t-test if and only if $A > 0$ for upper tailed test and $A < 0$ for lower tailed test. Let $\hat{\eta}_{ik,0.05}$ and $\hat{\eta}_{ik,0.95}$ denote the estimated 5^{th} and 95^{th} percentile of the approximated distribution of d_{ik}, and let $t_{ik,0.05}$ and $t_{ik,0.95}$ denote the 5^{th} and 95^{th} percentile of t distribution of two sample t-test of z_{1ikl} and z_{2ikl}. Based on the result of Zhang & Wang[12], we use the following procedure to decide the cutoff value, which will be used as the threshold value for d_{ik}.

In order to reduce the noise, we only keep the genes whose d_{ik} is more extreme than the critical values $cutoff\_up_{ik}$, $cutoff\_low_{ik}$. These genes are retained as potential informative

```
for each gene i do
    for each class k do
        if upper tailed test and $\hat{A} < 0$ then
            $cutoff\_up_{ik} \leftarrow t_{ik,0.95}$
            $cutoff\_low_{ik} \leftarrow t_{ik,0.05}$
        else if upper tailed test and $\hat{A} \geq 0$ then
            $cutoff\_up_{ik} \leftarrow \hat{\eta}_{ik,0.95}$
            $cutoff\_low_{ik} \leftarrow \hat{\eta}_{ik,0.05}$
        else if lower tailed test and $\hat{A} \geq 0$ then
            $cutoff\_up_{ik} \leftarrow t_{ik,0.95}$
            $cutoff\_low_{ik} \leftarrow t_{ik,0.05}$
        else if lower tailed test and $\hat{A} < 0$ then
            $cutoff\_up_{ik} \leftarrow \hat{\eta}_{ik,0.95}$
            $cutoff\_low_{ik} \leftarrow \hat{\eta}_{ik,0.05}$
```

genes. This is equivalent to set $d^*_{ik} = d_{ik}$ when $d_{ik} < cutoff\_low_{ik}$ or $d_{ik} > cutoff\_up_{ik}$; and set $d^*_{ik} = 0$, otherwise, i.e.

$$d^*_{ik} = d_{ik} I_{(d_{ik} < cutoff\_low_{ik} \ \ or \ \ d_{ik} > cutoff\_up_{ik})}.$$

The genes with $d^*_{ik} = 0$ are excluded from further analysis, and those genes with $d^*_{ik} \neq 0$ are very likely to have influence on the class label. Now we have obtained a new set of values d^*_{ik}, which will be used in the following soft thresholding procedure.

Next, we shrink d^*_{ik} further by a soft thresholding procedure. Let

$$\begin{aligned} d^{\#}_{ik} &= sgn(d^*_{ik})(|d^*_{ik}| - \Delta)_+ \\ &= sign(d_{ik})(|d_{ik}| - \Delta)_+(1_{d_{ik} > max(\Delta, cutoff\_up_{ik})} + 1_{d_{ik} < min(-\Delta, cutoff\_low_{ik})}), \end{aligned}$$

where Δ is a threshold value determined by 10-fold cross validation.

In the 10-fold cross-validation, each Δ value corresponds to a model, which has CV error and corresponding selected genes. The Δ value corresponding to the smallest CV error is the optimal thresholding value. Sometimes, the second smallest CV error is within one standard deviation of the smallest error, which means, the prediction performance has little difference with these two threshold values. In many cases, chasing after the threshold value can lead to too many variables retained in the model (which is a sign of overfit). Therefore we recommend the following: if the second smallest error is within one standard deviation of the smallest error and the number of selected genes for the second smallest CV error is less than the number of genes correspond to the smallest CV error, we choose the threshold value correspond to the second smallest error as the optimized threshold value. Otherwise, the optimal threshold is set to be the one with smallest training error.

With the selected optimal threshold value, the shrunken centroid for i^{th} gene and k^{th} class can be written as:

$$\bar{x}'_{ik} = \bar{x}_i + m_k(s_i + s_0)d^{\#}_{ik}.$$

This shrunken centroid is then used to compute the discriminant function for a new sample $x^* = (x^*_1, x^*_2, ..., x^*_p)$ belonging to *class k*:

$$\delta_k(x^*) = \sum_{i=1}^{p} \frac{(x^*_i - \overline{x}'_{ik})^2}{(s_i + s_0)^2} - 2\ log\ \pi_k$$

The first term is the $L-2$ norm of the studentized distance of the new sample x^* to the k^{th} shrunken centroid. If the distance is small, the new sample is more likely to belong to *class k*. The class $C(x^*)$ for this new sample is:

$$C(x^*) = \underset{k}{\operatorname{argmin}}\ \delta_k(x^*)$$

The π_k is the frequency of *class k*, where $\sum_{k=1}^{K} \pi_k = 1$. The $-2log\pi_k$ term included in the discriminant function will make sure the discriminant function give appropriate credit to unbalanced sample sizes. This is particularly important when none of the variables are informative to the class status and the sample sizes are severely unbalanced. For example, suppose 90% samples belong to class 1 and 10% belong to class 2 and no given variables are related to the class status. Then the $-2log\pi_k$ will be smaller for class 1 and bigger for class 2. This will make about 90% of samples being classified to class 1, which leads to misclassification error of around 10%. Without this term, random guessing will put 50% sample in class 1. That leads to at least 40% misclassification error.

We can apply softmax function to make this discriminant score a probability. The probability of the new sample x^* belongs to *class* k is:

$$\hat{p}_k(x^*) = \frac{e^{-\frac{1}{2}\delta_k(x^*)}}{\sum_{l=1}^{K} e^{-\frac{1}{2}\delta_l(x^*)}}$$

For each given new sample x^*, we can decide the class label with the largest probability $\hat{p}_k(x^*)$.

Chapter 3

A simulation study

In this chapter, we will present the result of a simulation study to compare the performance of TCF-PAM and PAM.

We generate a binary class dataset $(y_j, x_{ij}),\ i = 1, ..., p,\ j = 1, ..., n$, in which $y_j \in \{1, 2\}$ denotes the class label for the j^{th} observation and x_{ij} denotes the j^{th} observation of i^{th} variable. Let $n_1,\ n_2$ denote the sample size for classes 1 and 2, respectively. Denote $\lambda = n_1/n$, the sample proportion of observations from class 1.

Inspired by the fact that high throughput data are often skewed, we generate data using skewed distribution and symmetric distribution. In the first class, $x_{ij} = \epsilon_{ij} + 0.5 * \sqrt{2}$, where $\epsilon_{ij} \sim Gamma(shape = 0.08, rate = 0.2)$, for $i = 1, ..., 5$, and $x_{ij} \sim Normal(\mu = 0.4, \sigma^2 = 2)$, for $i = 6, ..., 1000$. That is, the first five variables in the first class have shifted gamma distribution while the rest of the variables are normally distributed. In the second class, all variables are independently generated from $Normal(\mu = 0.4, \sigma^2 = 2)$. hence, the first five variables are informative in that they determine the class label. The remaining variables are non-informative since they have no contribution to the class label. Different sample size ratios between the two populations may have an influence on the result, hence we set $\lambda = 0.1, 0.2, ..., 0.9$ and fix $n_1 = 250$. With fixed n_1, the sample size for the second class $n_2 = 250(\lambda^{-1} - 1)$ is a decreasing function of λ. Therefore smaller λ corresponds to bigger total sample size while large λ correspond to less total sample size. Since both small and large values of λ lead to unbalanced sample sizes, we can expect different performance depending on how many samples are allocated to the skewed population. The cases with λ close to 1 should be more difficult to classify than the case with other λ values.

The generated samples will be split into almost balanced two parts, one part for variable selection and training the models, and the other part for assessing the prediction accuracy. Our goal is to identify informative variables and reduce the selection of non-informative variables by the variable selection procedure described in Chap 2.

The result in Table 3.1 is based on 50 runs. In each run we randomly generate a new binary class dataset according to the data generation described in the beginning of Chap 3. The first six columns in the table are the average cross-validation error for the training data (CV error), prediction error for the test data (test error), and the prediction error if the modeling only have the first five variables included for the two methods (Oracle error). The last six columns show the average number of variables selected. TP (true positive) refers to the average number of selected informative variables, i.e. correctly identified variables among the first five. The column FP (false positive) gives the average number of selected non-informative variables, and the column Oracle shows the number of variables selected if the model only includes the first five variables, i.e, all the variables are informative and no other variables included. The standard error from the 50 runs is shown in parenthesis.

From Table 3.1, the cross-validation error and test error are similar for both methods which means there does not seem to have overfiting problems. To compare the prediction performance, we conduct two-sample one-sided t-test on prediction error of the two methods for each λ value. Based on the p-value given by the two-sample t test, we conclude that the prediction error of TCF-PAM is significantly lower than the prediction error of PAM when $\lambda = 0.1, 0.2, 0.3, 0.4, 0.6, 0.7, 0.8$. Highly significantly different performance was observed when $\lambda = 0.1$ (p-value=1.3e-11), $\lambda = 0.2$ (p-value=1.7e-15), $\lambda = 0.6$ (p-value =1.1e-19), $\lambda = 0.7$ (p-value=5.1e-11). In all these cases, the prediction performance of TCF-PAM is much better than that of PAM. The differences at $\lambda = 0.5$ and $\lambda = 0.9$ is not significant (at $\lambda = 0.5$, p-value=0.06; at $\lambda = 0.9$, p-value = 0.355).

In terms of selection of true informative (TP) variables, TCF-PAM has much better

	PAM			TCF			PAM			TCF		
	CV	test	Oracle	CV	test	Oracle	# selected variables			# selected variables		
λ	error	error	error	error	error	error	FP	TP	Oracle	FP	TP	Oracle
0.1	10	10	10	9.7	9.8	10	56.4	0.8	1	30.5	4.8	0.7
	(0)	(0)	(0)	(0)	(0)	(0)	(26.3)	(0.3)	(0.3)	(4.5)	(0.1)	(0.2)
0.2	19.8	20	19.9	19.1	19.2	19.9	293.8	2.6	3	13.9	5	1.7
	(0)	(0)	(0)	(0.1)	(0.1)	(0)	(53.1)	(0.4)	(0.3)	(2.5)	(0)	(0.3)
0.3	29	29.6	29.1	27.9	28.8	29.4	306.7	4.8	5	12.3	4.8	3
	(0.1)	(0.1)	(0.1)	(0.2)	(0.2)	(0.1)	(31.1)	(0.1)	(0)	(2.8)	(0.1)	(0.3)
0.4	36	37.5	36.7	33.6	35	38.8	262.1	5	5	48.1	4.9	3.2
	(0.3)	(0.2)	(0.2)	(0.5)	(0.4)	(0.2)	(28.8)	(0)	(0)	(4.4)	(0.1)	(0.2)
0.5	28.2	29.3	27.7	28.2	26.8	30	13.1	4.2	4.4	9	4.2	3.6
	(1)	(1.2)	(1.2)	(1)	(1.1)	(1.5)	(7.2)	(0.2)	(0.2)	(2.9)	(0.1)	(0.1)
0.6	35.6	36.7	31	28.5	28.2	29.3	203.2	4.9	5	18.5	4.5	3.8
	(0.4)	(0.3)	(0.4)	(0.6)	(0.6)	(0.7)	(24.8)	(0)	(0)	(2.3)	(0.1)	(0.1)
0.7	28.9	29.9	29.4	26.2	26	27.3	256.7	3.4	2.8	26.2	4	3
	(0.2)	(0.2)	(0.1)	(0.4)	(0.5)	(0.4)	(37)	(0.3)	(0.4)	(2.6)	(0.2)	(0.2)
0.8	19.8	19.9	19.9	18.6	18.9	18.7	60.3	0.8	0.4	24.5	2.6	2.2
	(0)	(0)	(0)	(0.3)	(0.2)	(0.3)	(21)	(0.3)	(0.2)	(2.6)	(0.2)	(0.3)
0.9	10.1	10.1	10.1	9.6	10.1	9.9	0	0	0	17.7	1.1	0.5
	(0)	(0)	(0)	(0.2)	(0.2)	(0.1)	(0)	(0)	(0)	(3.4)	(0.2)	(0.1)

Table 3.1: Result for simulation study. CV error is the average cross-validation error for the training data and test error is the prediction error for the test data. TP (true positive) shows the average number of correctly selected informative variables. The column FP (false positive) shows the average number of selected non-informative variables. The column Oracle error and Oracle give prediction error and number of variables selected if the model only included the informative variables. The standard error from the 100 runs is shown in parenthesis (rounded to keep only 1 significant digit).

performance when the dataset is heavily unbalanced, i.e, $\lambda = 0.1, 0.2, 0.8, 0.9$. In particular, when $\lambda = 0.1$ or 0.2, nearly all informative variables were selected by TCF. But PAM on average selected less than 1 informative variable when $\lambda = 0.1$ and selected less than 3 informative variables when $\lambda = 0.2$. When $\lambda = 0.9$, the dataset is heavily unbalanced but the sample size is small. PAM loses its ability of variable selection , and made its decision based on class proportion. TCF still can select informative variables.

In terms of the number of non-informative variables selected by mistake (FP), TCF-PAM can select less non-informative variables in all the situations except when $\lambda = 0.9$. In particular, for most λ values, the number of flase positives for PAM is often several hundred while that for TCF-PAM is mostly less than 30. TCF-PAM not only gives less false positive than PAM, but also has smaller fluctuation in its results. This can be seen from the smaller standard error of FP for TCF-PAM than that for PAM, which means TCF-PAM gives more stable results. When $\lambda = 0.9$, the FP for PAM is zero because PAM can not select any variable with small sample size.

In the performance of oracle situation, i.e, only the first five variables were provided to the TCF-PAM and PAM algorithms, we can see that PAM on oracle dataset seems to slightly outperform the PAM with all 1000 variables. On the other hand, the TCF-PAM applied to 1000 variables outperform its application to the oracle data. For the case with nearly balanced samples sizes ($\lambda = 0.4, 0.5, 0.6$), knowing the true informative variables in the oracle case helps PAM to slightly reduce the prediction error. On the other hand, the oracle information in the nearly balanced case did not help TCF-PAM in prediction error. For those moderate to severely unbalanced cases, the prediction errors for the TCF-PAM or PAM applied to the oracle data are comparable to those when they were applied to the data with 1000 variables. Such result of TCF-PAM showing similar predict error but with less number of selected variables tells that there is redundancy in the informative variables. Even though five variables clearly have different distributions for the two classes, some subset of them can provide the same information as all five variables.

Chapter 4

Applications to real data

In this section, we will use some public microarray data to compare the performance of TCF-PAM with PAM.

4.0.1 Application 1: Comparison of performance of two methods on 10 datasets

The summary of the datasets is in Table 4.1.

As discussed in previous section, we are interested in the test error and number of genes selected. A better method has smaller test error and possibly smaller number of genes selected. Since we use 10-fold cross-validation to decide the optimal threshold value, the

dataset	n.class	gene	ntrain	ntest	TrainClassSize	TestClassSize	Source
Leukemia1	3	7129	38	34	11,19,8	14,19,1	[13]
Leukemia2	3	12582	57	15	20,20,17	4,8,3	[14]
Lung1	3	7129	64	32	44,13,7	23,6,3	[15]
SRBCT	4	2308	63	20	8,23,12,20	3,6,6,5,5	[16]
Breast	5	9216	54	30	7,12,6,20,9	3,7,3,12,5	[17]
Lung2	5	12600	136	67	93,13,12,4,14	46,7,5,2,7	[18]
DLBCL	6	4062	58	30	7,4,7,30,6,4	3,2,4,16,3,2	[19]
Leukemia3	7	12558	215	112	28,52,9,18,42,14,52	15,27,6,9,22,6,27	[20]
Cancers	11	12533	100	74	8,12,11,11,10,6,9,8,9,6,10	14,12,1,1,1,5,6,18,16	[21]
GCM	14	16063	144	46	8,8,16,8,24,8,16,8,8,8,8,8,8,8	3,3,4,3,6,3,6,2,3,3,3,2,3,2	[4]

Table 4.1: Summary of datasets

random patitioning of data could lead to different test result. We run the random partition 100 times on each dataset. For each partition, we record the test error and the number of selected genes. Table 4.2 and Fig 4.1 summarize the result of applying the two methods to the Breast cancer dataset. Similiar summaries for other datasets are given in the appendix. The side by side barplot shows the number of errors in the horizontal axis and frequency out of 100 runs in the vertical axis. The label for each bar gives the average number of selected genes with median absolute deviation (MAD) in parenthesis. A method with better performance is expected to have more occurrence in the left side of the plot. The five number summary table shows numerical summary of the test error and the number of genes selected by PAM and TCF-PAM.

	TCF		PAM	
	test Er	# genes	test Er	# genes
Min.	0.00	264	1.00	392
1st Qu.	1.00	598	3.00	4819
Median	1.00	933	3.00	4819
Mean	1.32	1793	2.89	4297
3rd Qu.	1.00	2427	3.00	4819
Max.	5.00	7504	5.00	8960

Table 4.2: Five Number Summary of the # of misclassification subjects among 30 test samples and the number of genes selected by PAM and TCF-PAM.

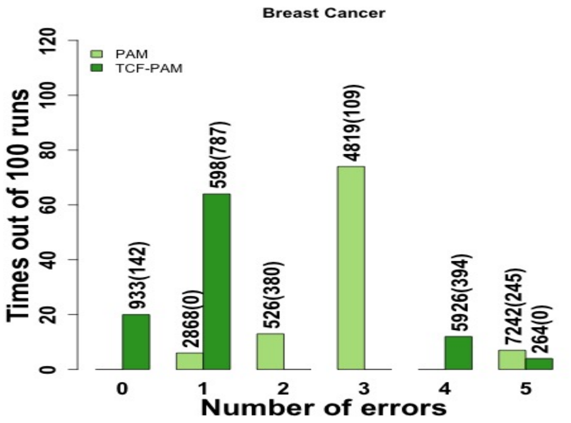

Figure 4.1: Side by side barplot of the number of errors versus frequency out of 100 runs. The label for each bar gives the average number of selected genes with MAD in parenthesis.

From Table 4.2, we can see that more than 75% times out of 100 runs, the TCF-PAM has prediction error less than 1/30 while more than 75% prediction errors being more than 3/30 for PAM. The number of genes selected by TCF-PAM is less than 1000 in 50% of the runs, while PAM slected more than 4800 genes in more than 75 runs. It is apparent from the side by side barplot in Fig 4.1 that most of the errors for TCF-PAM is 0 or 1 out of 30 subjects in the test samples, and most of the errors for PAM is 3 out of 30. TCF-PAM will need on average 933 genes to reach 0 error or 598 genes to reach 1 error while PAM's best performance requires on average of 2868 genes to reach 1 error. Majority of time, PAM selected on average 4819 genes and the error is 3 out of 30 subjects.

To compare the performance of the two methods on all 10 datasets, we model the probability of making a correct prediction from each method using the 100 runs. The higher this probability for a method, the better the method. We use a binary logistic regression model to estimate this parameter. The logistic regression model has a factor 'method' with two levels, TCF-PAM and PAM, with the response being whether a correct prediction was made. The log odds of successful prediciton is given by

$$\log(\frac{p_{success,i,j}}{1-p_{success,i,j}}) = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij}, i = TCF - PAM \ or \ PAM, j = 1, ..., 10. \quad (4.0.1)$$

where $p_{success,i,j}$ is the probability of making a correct prediction using i_{th} method on j_{th} dataset, μ is the intercept, α_1 and α_2 are the effects of TCF-PAM and PAM, respectively; β_j is the effect of the jth datasets $j = 1, 2, ..., 10$; $(\alpha\beta)_{ij}$ is the interaction effect of method and dataset. To access the interaction effect, a deviance test was conducted by comparing the model in (4.0.1) with the following model:

$$\log(\frac{p_{success,i,j}}{1-p_{success,i,j}}) = \mu + \alpha_i + \beta_j, \ i = TCF - PAM \ or \ PAM, \ j = 1, ..., 10. \quad (4.0.2)$$

The chisquare test of model (4.0.2) against model (4.0.1) gives p-value = 2.2e-16, which gives us strong evidence that the two methods have different performance on different datasets. This conlusion matches the result of our simulation study that TCF-PAM has better performance when the dataset is unbalanced. Since the perfomance varies depending on the dataset, we next compare the performance of TCF-PAM and PAM on each dataset using the model:

$$\log(\frac{p_{success,i}}{1-p_{success,i}}) = \mu + \alpha_i, \ i = TCF \ or \ PAM.$$

The deviance test of $H_0 : \alpha_i = 0$ yields the p-value shown in Table 4.3. In Table 4.3, log odds ratio refers to the log odds of TCF-PAM over PAM for predicting successfully versus incorrectly, i.e. estimate of $\alpha_1 - \alpha_2$, where α_1 is the effect of TCF-PAM and α_2 is the effect

						PAM		TCF-PAM	
Group	dataset	adjusted p-value	log odds ratio	odds ratio	1/odds ratio	median	mean	median	mean
1	Cancers	1				1980	1952	3544	4453
	DLBCL	1				3808	3677	3835	3762
	Lung1	1				7	73	1	331
	SRBCT	1				136	140	57	84
2	Lung2	0.0036	-0.52	0.6	1.7	2674	2650	1042	1775
	Leukemia1	1.2e-33	-1.3	0.27	3.7	63	159	28	978
	Leukemia3	7.8e-86	-0.96	0.38	2.6	10342	9679	3893	3915
3	Leukemia2	2.5e-06	0.65	1.9	0.52	7959	5816	24	275
	GCM	2.7e-05	0.22	1.2	0.8	2395.5	2520	5811	6630
	Breast	1e-13	0.84	2.3	0.43	4819	4297	933	1793

Table 4.3: Result of comparing the two methods for 10 datasets. Adjusted p-value is the p-value adjusted with Bonfferoni correction of multiple comparisons. The column of Log odds ratio shows the log odds of effect of TCF over effect of PAM. The last four columns show the median and mean number of genes selected by the two methods.

of PAM. This is because the odds ratio is

$$\frac{\frac{p_{TCF}}{1-p_{TCF}}}{\frac{p_{PAM}}{1-p_{PAM}}} = \exp\{\alpha_{TCF} - \alpha_{PAM}\}.$$

For example, the odds ratio of successful prediction for TCF-PAM over PAM is 2.3 for the Breast cancer data, which means that odds of making a correct prediction for TCF-PAM is 2.3 times of that of PAM.

The first group in Table 4.3 list the datasets on which the two methods did not show significant difference in the probability of making correct predictions. Since prediction accuracy is not significantly different in these cases, the number of genes used in the prediction tells us which method is more powerful. TCF-PAM and PAM have similar performance for the dataset DLBCL since they use similar number of genes. PAM is better for the dataset Cancers while TCF-PAM is better for datasets SRBCT. For the dataset Lung1, the median and mean for TCF-PAM are quiet different, which means the number of genes selected are skewed from different runs of each algorithm. TCF-PAM can achieve relatively small prediction error with less number of genes in most cases (more than half). But in some situations it can give relatively high prediction error with large number of selected genes because of bad spliting in cross-validation. Then, intuitively, we may choose to resplit the dataset and re-run the program.

The second group in Table 4.3 shows the datasets on which PAM is more likely to make a correct prediction. With the frequency of occurrence out of 100 runs and number of genes selected plots shown in the Appendix, we describe the results for several datasets. For the dataset Lung2, TCF-PAM misclassifies all the test samples in class 'SQUA' to class 'ADEN' and correctly classify all samples belonging to 'ADEN', while PAM correctly predicts 'SQUA' samples but makes mistakes on 'ADEN'. While the class 'ADEN' has a large proportion of samples (93 samples) in the training dataset, TCF-PAM enlarges the effect of this class. Similar pattern happens for the dataset Leukemia1, TCF is more likely to predict a new sample to the class with large class size in the training data. The class 'T_cell' (8 samples in training data) is more likely to be classified to the class 'B_Cell' (19 samples in training data) by TCF-PAM. Considering that the class 'T_Cell' and class 'b_Cell' in the dataset Leukemia1 belong to same class in some studies[13], the two classes are similar. For dataset Leukemia3, TCF-PAM loses the power for class 'BCR' (9 samples in training data). Aimed to choose informative genes, TCF-PAM loses some power for the class with small training class size. When we compare the genes selected by the two methods, TCF-PAM aims to reduce the number of genes while PAM selects too many genes which may not give much information about the class status.

The third group of Table 4.3 is the dataset on which TCF-PAM has significantly larger probability of making a correct prediction. For the dataset Leukemia2, TCF-PAM and PAM had similar range in the number of misclassification, but TCF-PAM has higher frequency of smaller errors, which leads to better prediction accuracy in TCF-PAM. And the number of selected genes is drastically reduced by TCF-PAM. There are 14 classes for the dataset GCM, which makes classification difficult for most classifier. For all the 100 runs on GCM dataset (A.1), TCF-PAM has smaller prediction error than that of PAM. For the Breast cancer dataset, as discussed at the beginning of this section, TCF-PAM is 2.3 times more likely to give small prediction error using a small number of genes than PAM.

	LOOCV error	# of genes selected
mRMR-IFS	0.105	117
PAM	0.042	28
TCF	0.053	16

Table 4.4: Result of three methods on gene expression in nontumoral liver tissue and recurrence-free survival in hepatitis C virus-positive HCC dataset (GSE 17856)

4.0.2 Application 2: Comparison to mRMR-IFS on GSE17856 through leave-one-out CV

We also apply the two methods on the dataset GSE17856 from NCBI. This dataset has 95 samples from 2 classes (43 tumor, 52 non-tumor) and 25073 genes.

We compare the performance of TCF-PAM, PAM and the mRMR-IFS strategy proposed by Gui et al.[22]. The method mRMR-IFS first trains all the samples to get the rank of each gene, then it selects a subset of top ranked genes as the variables for prediction. The prediction result for each sample is based on nereast neighbors. The subset of genes with the smallest error is selected by this method. Gui et al. reported leave-one-out cross validation error. To compare the results with TCF-PAM and PAM, we also apply these two methods on the dataset GSE17856 to get the LOOCV error. The LOOCV procedure is as follows: Leaving out a single sample from the dataset as the validation data, we train the models with the remaining samples. This is repeated such that all the samples in the dataset is used once as the validation data. This is the same as a K-fold cross-validation with K being equal to the number of samples in the dataset. The total number of misclassifications is the LOOCV error.

Summary of the results is given in Table 4.4. The first column is the LOOCV error for each method. The second column is the number of genes selected. The LOOCV error of TCF-PAM and PAM is smaller than the traning error of mRMR-IFS which means TCF-PAM and PAM outperform mRMR-IFS. The LOOCV error of TCF-PAM has 1% more error than PAM, but the number of genes used in TCF-PAM is about half of those by PAM.

The datasets GSE 17856 also gives the GO-ID and its corresponding functions. These selected genes potentially have influence on tumor. We show the functions of the 16 genes selected by TCF in Table 4.5.

GenBank Accession #	Function
NM014652	(protein binding)(nucleus)(cytoplasm)(protein import into nucleus) (intracellular protein transport)(protein transporter activity)
NM025082	(M phase of mitotic cell cycle)(mitotic prometaphase) (mitotic cell cycle)(condensed chromosome kinetochore) (DNA binding)(nucleus)(chromosome)(cytosol)
NM001033112	(translation repressor activity)(regulation of translation)(cytoplasm) (negative regulation of translational initiation)(protein binding)
NM022748	(protein binding)(focal adhesion)(lung alveolus development) (positive regulation of cell proliferation)(cell migration) (cell junction)
NM032257	(binding)(intracellular)(zinc ion binding)(metal ion binding)
NM014789	(DNA binding)(intracellular)(nucleus)(metal ion binding) (regulation of transcription, DNA-dependent)(zinc ion binding)
NM032689	(DNA binding)(intracellular)(nucleus)(metal ion binding) (regulation of transcription, DNA-dependent)(zinc ion binding)
NM005586	(negative regulation of transcription from RNA polymerase II promoter) (nucleus)(cytoplasm)(activation of JUN kinase activity) (embryo development)(dorsal/ventral axis specification) (negative regulation of Wnt receptor signaling pathway) (cytoplasmic sequestering of transcription factor)(protein binding) (negative regulation of DNA binding)(cell differentiation) (transcription factor binding)(trophoblast giant cell differentiation) (embryonic skeletal system morphogenesis)
NM175075	(molecular function)(cellular component)(biological process)
NM003124	(aldo-keto reductase (NADP) activity)(sepiapterin reductase activity) (nucleolus)(cytoplasm)(tetrahydrobiopterin biosynthetic process) (nitric oxide biosynthetic process)(oxidoreductase activity) (NADP binding)(oxidation-reduction process)
NM024653	(double-stranded RNA binding)(protein kinase inhibitor activity) (nucleus)(nucleolus)(negative regulation of protein kinase activity) (protein kinase binding)(negative regulation of phosphorylation)
NR045217	

Table 4.5: Summary of gene functions

Conclusion

In this report, we presented the TCF-PAM, an improved version of PAM method. The PAM method assumed that all the genes are identically distributed which may not be the case in real applications. Through introducing the TCF-PAM test on each variable(gene), we derived variable specific thresholding parameter. This is achieved through seperate higher order expansion to approximate the quantiles of the comparison statistic. Cross validation was further used to select the final set of variables with training data. Numerical studies show that our method achieve similar prediction error using less variables and therefore less complex models. Our method significantly reduces false positive rate compared to the original PAM algorithm.

In applying Naive Bayes algorithm as the starting point, we also assumed that all the variables(genes) are independent so that the variable specific thresholding parameter can be selected independently. If the variables(genes) are not independent, we recommend to first use sparse principle component analysis to convert the variables into linearly uncorrelated features before applying the TCF-PAM algorithm.

Bibliography

[1] J. G. Thomas, J. M. Olson, S. J. Tapscott, and L. P. Zhao. An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles. *Genome Res*, 11(7):1227–36, 2001.

[2] C. A. Tsai, Y. J. Chen, and J. J. Chen. Testing for differentially expressed genes with microarray data. *Nucl. Acids Res*, 31:e52, 2003.

[3] A. Antoniadis, S. Lambert-Lacroix, and F. Leblanc. Effective dimension reduction methods for tumor classification using gene expression data. *BMC Bioinformatics*, 19: 563–570, 2003.

[4] S. Ramaswamy, P. Tamayo, R. Rifkin, S. Mukherjee, C. H. Yeang, M. Angelo, C. Ladd, M. Reich, E. Latulippe, J. P. Mesirov, T. Poggio, W. Gerald, M. Loda, E. S. Lander, and T. R. Golub. Multiclass cancer diagnosis using tumor gene expression signatures. *Proc Natl Acad Sci USA*, 98:15149–15154, 2001.

[5] A. C. Tan, D. Q. Naiman, L. Xu, R. L. Winslow, and D. Geman. Simple decision rules for classifying human cancers from gene expression profiles. *BMC Bioinformatics*, 21(20):3896–3904, 2005.

[6] R. Tibshirani, T. Hastie, B. Narasimhan, and G. Chu. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proc Natl Acad Sci USA*, 99:6567 – 6572, 2002.

[7] S. Ray, M. Britschgi, C. Herbert, Y. Takeda-Uchimura, A. Boxer, K. Blennow, L. F Friedman, D. R Galasko, M. Jutel, A. Karydas, J. A Kaye, J. Leszek, B. L Miller, L. Minthon, J. F Quinn, G. D Rabinovici, W. H Robinson, M. N Sabbagh, Y. T So,

D Larry Sparks, M. Tabaton, J. Tinklenberg, J. A Yesavage, R. Tibshirani, and T. Wyss-Coray. Classification and prediction of clinical alzheimer?s diagnosis based on plasma signaling proteins. *Nat Med*, 13:1359 – 1362, 2007.

[8] E. Bair and R. Tibshirani. Semi-supervised methods to predict patient survival from gene expression data. *PLoS Biol*, 2(4):E108–10.1371/journal.pbio.0020108., 2004.

[9] E. Cornish and R. Fisher. Moments and cumulants in the specification of distributions. *Int Stat Rev*, 5:307–320, 1938.

[10] X. Zhou and D. Philip. Nonparametric confidence intervals for the one- and two-sample problems. *Biostatistics*, 6:187–200, 2005.

[11] J. Xu, X. Cui, and A. K. Gupta. Improved statistics for contrasting means of two samples under non-normality. *Br. J. Math. Stat. Psychol.*, 62:21–40, 2009.

[12] H. Zhang and H. Wang. New two-sample tests with heterogeneous variance and their theoretical power. *Manuscript in progress.*

[13] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531–537, 1999.

[14] D. G. Beer, S. L. Kardia, C. C. Huang, T. J. Giordano, A. M. Levin, D. E. Misek, L. Lin, G. Chen, T. G. Gharib, D. G. Thomas, M. L. Lizyness, R. Kuick, S. Hayasaka, J. M. Taylor, M. D. Iannettoni, M. B. Orringer, and S. Hanash. Gene-expression profiles predict survival of patients with lung adenocarcinoma. *Nat Med*, 8:816–824, 2002.

[15] S. A. Armstrong, J. E. Staunton, L. B. Silverman, R. Pieters, M. L. den Boer, M. D. Minden, S. E. Sallan, E. S. Lander, T. R. Golub, and S. J. Korsmeyer. Mll translocations specify a distinct gene expression profile that distinguishes a unique leukemia. *Nat Genet*, 30:41–47, 2002.

[16] J. Khan, J. S. Wei, M. Ringnr, 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. *Nat Med*, 7:673–679, 2001.

[17] C. M. Perou, T. Srlie, M. B. Eisen, M. van de Rijn, S. S. Jeffrey, C. A. Rees, J. R. Pollack, D. T. Ross, H. Johnsen, L. A. Akslen, O. Fluge, A. Pergamenschikov, C. Williams, S. X. Zhu, P. E. Lnning, A. L. Brresen-Dale, P. O. Brown, and D. Botstein. Molecular portraits of human breast tumours. *Nature*, 406:747–752, 2000.

[18] 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. Meyerson. Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. *Proc Natl Acad Sci USA*, 98:13790–13795, 2001.

[19] A. A. Alizadeh, M. B. Eisen, R. E. Davis, C. Ma, I. S. Lossos, A. Rosenwald, J. C. Boldrick, H. Sabet, T. Tran, X. Yu, J. I. Powell, L. Yang, G. E. Marti, T. Moore, J. Jr. Hudson, L. Lu, D. B. Lewis, R. Tibshirani, G. Sherlock, W. C. Chan, T. C. Greiner, D. D. Weisenburger, J. O. Armitage, R. Warnke, R. Levy, W. Wilson, M. R. Grever, J. C. Byrd, D. Botstein, P. O. Brown, and L. M. Staudt. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, 2000.

[20] E. J. Yeoh, M. E. Ross, S. A. Shurtleff, W. K. Williams, D. Patel, R. Mahfouz, F. G. Behm, S. C. Raimondi, M. V. Relling, A. Patel, C. Cheng, D. Campana, D. Wilkins, X. Zhou, J. Li, H. Liu, C. H. Pui, W. E. Evans, C. Naeve, L. Wong, and J. R. Downing. Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling. *Cancer Cell*, 1:133–143, 2002.

[21] A. I. Su, J. B. Welsh, L. M. Sapinoso, S. G. Kern, P. Dimitrov, H. Lapp, P. G. Schultz, S. M. Powell, C. A. Moskaluk, H. F. Frierson, and G. M. Hampton. Molecular classi-

fication of human carcinomas by use of gene expression signatures. *Cancer Res.*, 61: 7388–7393, 2001.

[22] T. Gui, X. Dong, R. Li, Y. Li, and Z. Wang. Identification of hepatocellular carcinoma-related genes with a machine learning and network analysis. *J Comput Biol*, 22(1): 63–71, 2015.

Appendix A

Appendix: Tables, Figures, and further description of the results for application 1

We list the detailed result from section 4.1 in Appendix.

We combine two kinds of plots in Table A.1, A.2, A.3, in which the left column is the plot of frequency out of 100 runs versus number of errors, and the right column is the square root of the median number of genes selected versus number of errors. For the left column, the better method will have higher bar for small errors which means the method is more likely to give small error, and lower bar for large errors which means the method is less likely to give large error. For the right column, because that the number of genes selected has a large range, we take square root of the median number of genes selected so that it is easy to compare different methods. And the error bar is the square root of median plus and minus the square root of MAD(mean absolute deviation) of the number of genes selected. The MAD is based on formula

$$\text{MAD} = \textit{constant} * \text{median}\,(\ |X_i - \text{median}(X)|\ )\,,$$

Where the constant use the default value 1.4826 in R.

The better method will have lower bar for al the possible number of errors and shorter error bar. If we can get same number of errors with less number of genes, these genes selected are more informative.

PAM performs better than TCF-PAM for datasets Leukemia1 and Lung2. It is clear that PAM gives just 1 error more than 95 times out of 100 runs for the Leukemia1 dataset. The number genes used for 1 error is 47 by PAM, which is a small number compared to 1108 used by TCF-PAM. For the dataset Lung2, PAM gives 0 error about half times when TCF-PAM will give at least 1 error.

The results for dataset SRBCT and DLBCL seems comparable from the two plots. They have a similar frequency for each possible number of errors. But TCF-PAM uses fewer genes to give 5 errors than the number of genes used by PAM for the dataset SRBCT.

PAM has pretty consistent results for the Cancers dataset. The prediction errors are always 9 while the number of selected genes varies each time. However, TCF-PAM can achieve 7 errors with fewer genes used. Although TCF-PAM may give 17 errors with almost all the genes included, it is more likely to give the less error result because about three fourths of the time TCF-PAM had 7 errors.

TCF-PAM outperforms PAM on the GCM dataset. Because the errors given by TCF-PAM is less than the errors given by PAM all the time. Although PAM selects fewer genes, low prediction accuracy means the genes selected are not informative or there is a lack of fit in the PAM model. TCF-PAM also outperforms PAM on the Breast cancer dataset. TCF-PAM has zero or one error most of times and with less number of genes selected, while PAM has three errors most of the times.

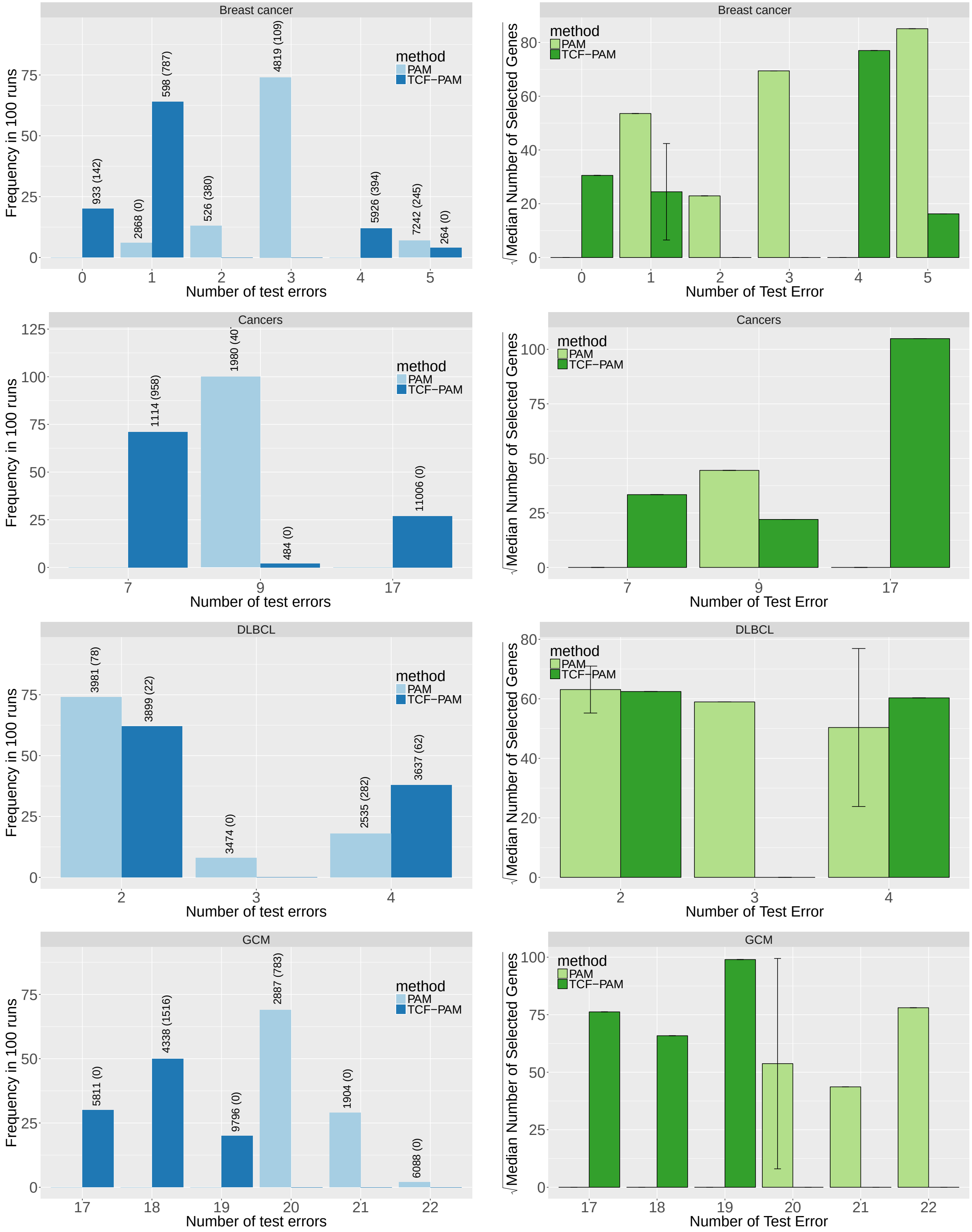

Table A.1: Frequency of occurrence out of 100 runs and number of genes selected. The left panel of each row is side by side frequency plot. The label on the bar is the median and median absolute deviation (MAD) of number of genes selected. The right panels are side by side plot of the number of genes. The error bar shows square root of MAD of number of genes selected.

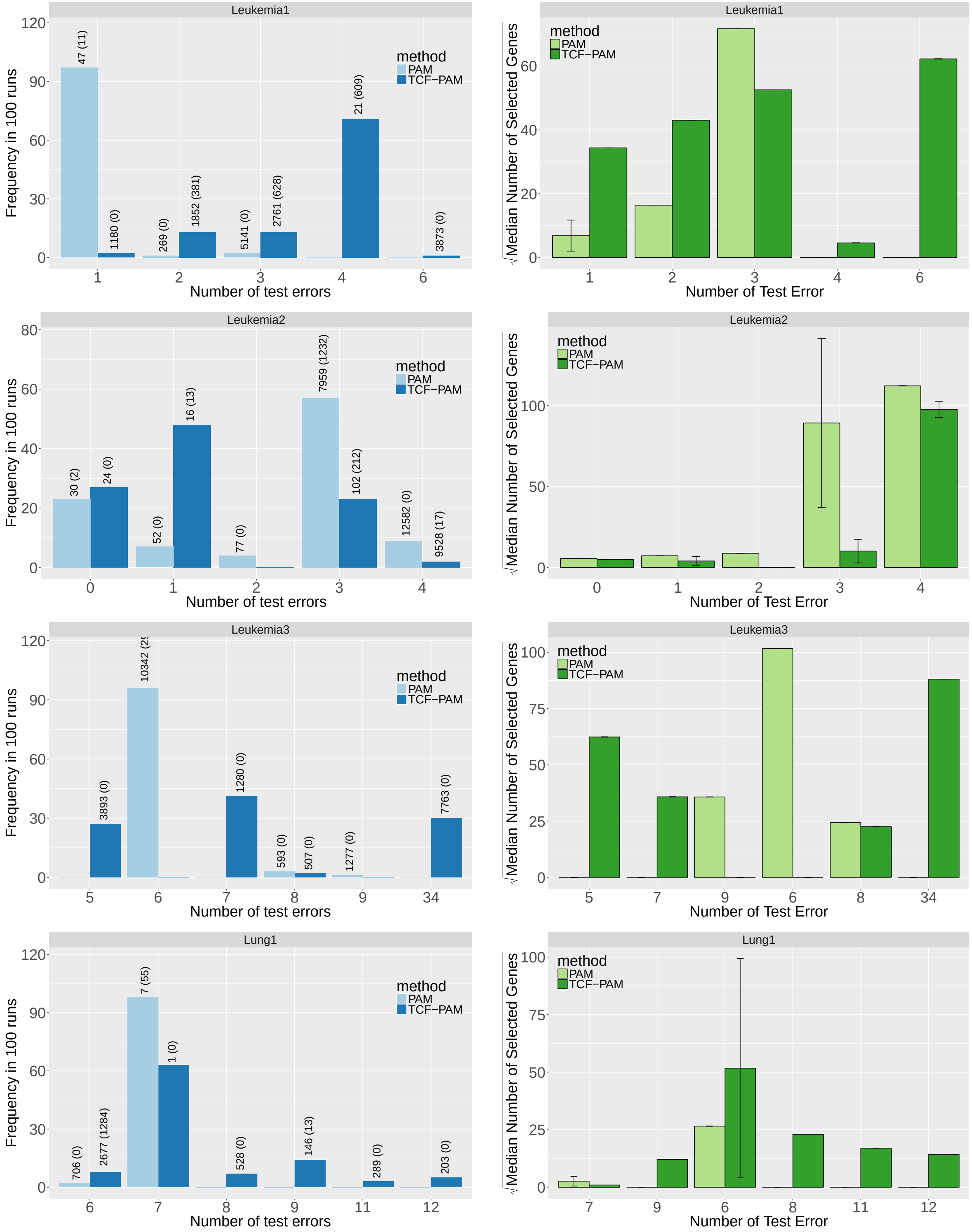

Table A.2: Frequency of occurrence out of 100 runs and number of genes selected. The left panel of each row is side by side frequency plot. The label on the bar is the median and median absolute deviation (MAD) of number of genes selected. The right panels are side by side plot of the number of genes. The error bar shows square root of MAD of number of genes selected.

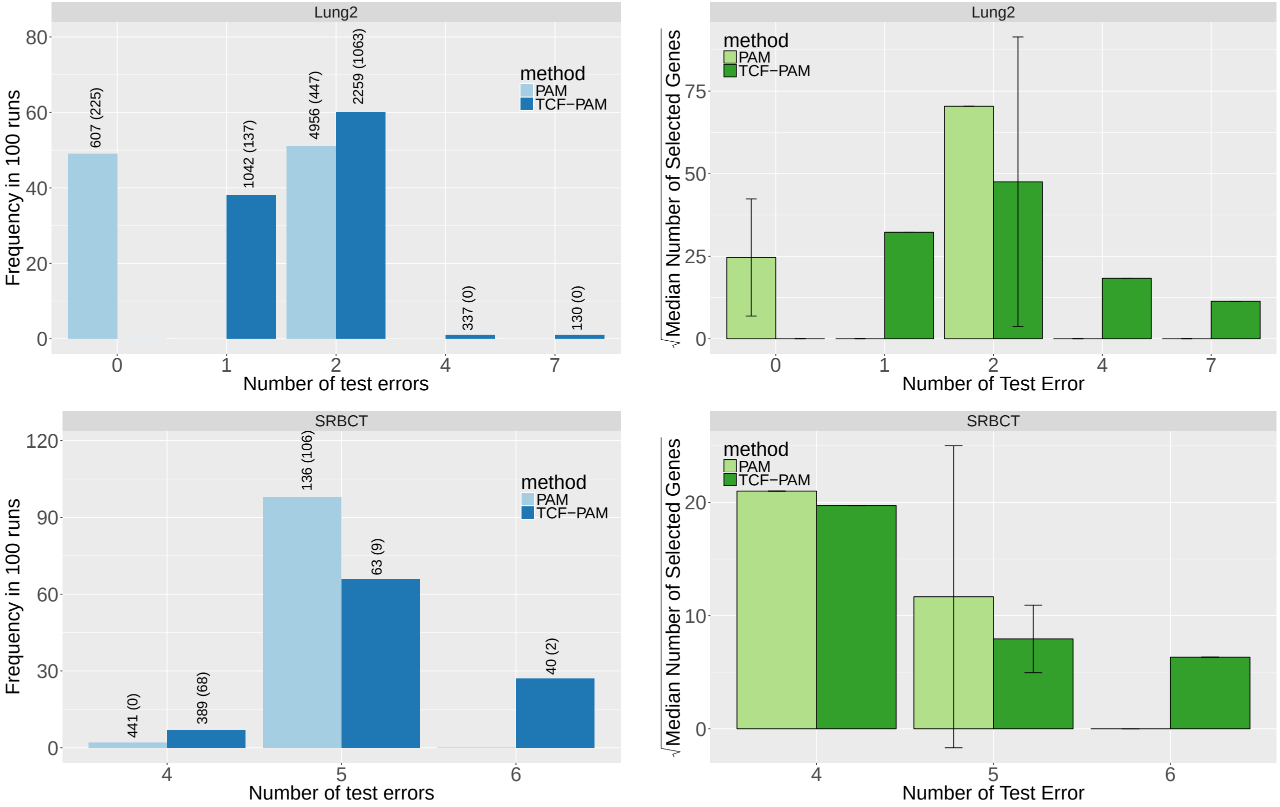

Table A.3: Frequency of occurrence out of 100 runs and number of genes selected. The left panel of each row is side by side frequency plot. The label on the bar is the median and median absolute deviation (MAD) of number of genes selected. The right panels are side by side plot of the number of genes. The error bar shows square root of MAD of number of genes selected.