STATISTICAL METHODS FOR GENE SELECTION USING DIFFERENTIAL GENE EXPRESSION AND BUILDING GENE CO-EXPRESSION NETWORKS

BY ZHAOYU LUO

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Statistis

Written under the direction of

Professor Javier Cabrera

and approved by

_____

_____

_____

_____

New Brunswick, New Jersey

October, 2009

ABSTRACT OF THE DISSERTATION

Statistical Methods for Gene Selection Using Differential Gene Expression and Building Gene Co-expression Networks

by Zhaoyu Luo

Dissertation Director: Professor Javier Cabrera

This thesis investigates three most challenging statistical problems that relate to three important stages of the pipeline of DNA microarray data analysis which are identification of differentially expressed genes, determination of sample size based on specified power, desired fold change and given error rate, and construction of gene co-expression network. At the center of these methods is a new version of the Stochastic Approximation methodology that works for distribution functions. The method is applied to estimation problems in the conditional-t procedure (Amaratunga and Cabrera (2003)) and in the estimation of the covariance matrix. The new covariance estimates are applied to the estimation of gene co-expression network (Zhang and Hovarth(2005)). In both cases the new method results in substantial improvement in performance. This is shown in several simulations that are presented throughout the thesis. In addition we show examples from real applications to illustrate the main results.

Preface

DNA Microarray is the most widely used technology in biomedical research to investigate the expression patterns of thousands of genes simultaneously. It is a powerful tool for biologists to explore the world of genes. At the same time, it imposes challenges for statisticians to analyze it because of its high dimension but small sample size. We need a set of new methodologies to deal with microarray data since traditional ones, which usually follow large sample principles, are lack of power.

We proposed improved conditional t test to detect differentially expressed genes, which is an important problem in microarray data analysis. Moreover, we developed a new approach to determine the sample size needed to gain a specific power and to satisfy some other conditions taking consideration of correlation among genes. We also built up a weighted gene co-expression network to explore the graphic information of genes. The use of these methodologies are not limited to microarray data and small sample size data. They could be applied to data with large sample size too. We proposed Stochastic Approximation for distribution (S.A.D) which were used in all three methods.

The outline of this thesis is as follows: Chapter 1 gives an introduction to microarray experiment and a summary of statistical issues related to microarray data analysis. We also review some work done in the literature. Since Stochastic Approximation for distribution (S.A.D) were used in all three methods in the thesis, the details of it is described in Chapter 2. In Chapter 3, improved conditional t test is introduced in detail. Comparison of various methods on simulated data and real data shows the superioity of our approach. We talk about our

methodology of sample size calculation in Chapter 4. We make use of estimated covariance matrix of genes to do simulation and gain more power. This approach works well in real data, especially for highly correlated gene groups. In Chapter 5, We propose a method to construct a gene co-expression network and apply our method to simulated data and a cancer data set. Finally, Chapter 6 summarizes our work and discusses some open questions.

Acknowledgements

First and foremost, I owe a great debt to my thesis advisor, Dr Javier Cabrera, Director of Institute of Biostatistics, Rutgers Uniersity. Every bit of progress I make, he is an essenticial part of it. My thesis research can not be done without his immensely valuable support,encouragement and advice.

I am also grateful to Dr Cun-Hui Zhang, former graduate director of statistics deprtment, Rutgers University. With his generous help, I bypassed one and another obstacles in both my study and in my life.

Special thanks go to my committee members.

I have benifitted from time spent with my colleages in Merck. They provided me with a wonderful working enviroment. I would especially thank Dr.Xiang Yu. His wonderful suggestions helped a lot on my research and I have learned a lot from him in the process of discussing problems.

Also I would like to thank Dr.Donghui Zhang and Dr.Li Liu. Working with them in Sanfi-aventis, I gained good ideas.

last, I am deeply indebted to my husband. He sacrificed his own time to support my study and our family.

Dedication

To my parents in China

Table of Contents

List of Tables

List of Figures

Chapter 1
Introduction

1.1 DNA Microarray Technology

The emergence of high-throughput experimental technologies has begun a new period of molecular biology. Tedious "one gene per experiment" paradigm is no longer a headache of biologists because DNA microarray, the most widely used form of this technology, allows biologists to monitor the expression profiles of a large number of genes at the same time and enables biologists to study how genes function jointly under a specific condition or under different conditions. Microarray technology has brought about great opportunities in functional genomics studies. It is a powerful tool to help scientists find out which pathway cause a disease or affect responses to treatment. For example, using microarrays Alizadeh et al.(2000) identified two molecularly distinct forms of diffuse large B-cell lymphoma (the most common subtype of non-hodgkin's lymphoma) which had gene expression patterns indicative of different stages of lymphoma. They showed that patients with one type of expressed genes had a significantly better overall survival than those with the other type of expressed genes.

This section gives the basic concepts of modern molecular biology and a typical protocol of an microarray experiment.

1.1.1 Biological background

Each gene, either by itself or in combination with some other genes,occupies a spot on a chromosome and determines a characteristic in an organism. Genes are made of deoxyribonucleic acid(DNA). A DNA molecule consists of two long strands wound tightly around each other in a spiral structure, which resembles a twisted ladder. The sides of DNA ladder are made of sugar and phosphate and the rungs of DNA ladder are made of bases. There are four kinds of bases: adenine(A),guanine(G),cytidine(C) and thymidine(T). The sequences of bases along each of the two strands of DNA are complementary to each other, following the complementary base-paring rules: A coupling to T and G coupling to C.

The genetic information can be transmitted from DNA to protein during gene expression via the process DNA $\rightarrow$ mRNA $\rightarrow$ protein, which means that the protein-coding instructions from a gene are transmitted indirectly through messenger ribonucleic acid (mRNA) is a molecule like a single strand of DNA except that in its base, uracil(U) replaces thymine(T).

Two DNA strands (or one DNA strand and one mRNA strand) whose sequences are complementary to each other will hybridize to form a single double-stranded DNA molecule. Even when the sequences on the two strands are not completely complementary to each other, if they share enough similarity, they may still form a DNA module with part of bases pairing. This property is exploited in hybridization assays. In these assays a probe consisting of a homogenous sample of single-stranded DNA modlecules of known sequence is prepared and labeled. A heterogeneous mixture of single-stranded DNA molecules of unknown composition is challenged by the probe. DNA sequences that are complemetary to the probe can be identified since the probe will hybridize only to these sequences. Among various types of hybridization, Northern blotting is the most commonly used one to detect gene expression levels. In Northern blotting, the target mRNA is extracted and transferrred to the surface of a solid support,e.g.,

a nylon filter. DNA microarrays can be viewed as a massively parallel version of Northern blotting.

Besides hybridization assays, there are serveral laborratory techniques that play an great role in microarray experiments. Polymerase chain reaction (PCR) is a rapid procedure for generating multiple copies of any fragment of DNA. Reverse transcription is a prcedure for reversing the process of transcription. It isolates mRNA, which is unstable and is easily degraded, and using it to synthesize a complementary DNA (cDNA) strand, which is stable and is not easily degraded. The cDNA generated by reverse transcription can be amplified by PCR, which is called reverse transcriptase polymerase chain reaction (RT-PCR).

1.1.2 Experiment procedures

In this subsection, I will describe the basic procedure of a typical microarray exprement. There are five steps:

1. Preparing the microarray: In this step, a drop of each known purified single-stranded DNA in the collection is robotically spotted to a specially prepared glass slide, which makes the DNA microarray for the experiment. The DNA spotted on the microarry are loned copies of cDNA, amplified by PCR, corresponding to whole or part of a fully sequenced gene. It can be either cDNA or ologonucleotides. In the former case, the microarry is called a cDNA microarray while in the latter case, the microarray is called an oligonucleotide array.

2. Preparing the labeled samples: After mRNA molecules are extracted from sample tissues, they are immediately reverse-transcribed into more stable cDNAs (for cDNA microarrays) or cRNAs (for oligonucleotide arrays). Then the sample is labeled with a reporter mocecule that flags their presence. The reporters currently used in microarray experiments are fluorescent dyes, called fluors or fluorophores.

3. Hybridizing the labeled samples to the microarray: The labeled sample is poured onto the microarray and allowed to diffuse uniformly all over it. Then it

is sealed in a hybridzation chamber and incubated at a specific temperature for enough time to allow the hybridization reactions to complete. Last, it is removed from the hybridzation chamber and thoroughly washed to eliminate any loose probes.

4. Scanning the microarray: The microarray is scanned to determine the amount of probes is bound to each spot. The probes are labeled with fluorescent reporter molecules which emit detectable light when stimulated by a laser. The emitted light is captured by a detector, either a charge-coupled device or a confocal microscope, that records its intensity. Spots with more bound probes will have more reporters and will therefore fluoresce more intensely.

5. Interpreting the scanned image: The end product of a microarray experiment is a scanned array image. The image will be converted into spot intensity measurements by image-processing software. High intensity spot means that DNA in that spot corresponds to some mRNA in sample and low intensity spot means that no mRNA in sample corresponds to the DNA at that spot.

1.2 Statistical issues in microarray data analysis

The raw data from a DNA microarray experiment is a series of scanned images of microarrays,one image per channel. The general plan for analyzing this data including converting these images into quantitative data, preprocessing the data and applying appropriate data analysis technques. Due to the extremely high dimension of microarray data and the many scources of variation intrduced during microarray experiments, either standard methods has to be tailored for use with microarray data or an entirely fresh set of tools has to be developed specially to handle such data. In this section, statistical tools applied to microarray data are outlined.

1.2.1 Processing the Scanned Image

The end product of a microarray experiment is a scanned image and the image have to be converted into spot intensities for the use of analysis. Three steps are involved in quantifying a scanned image. The first step, gridding, is to define the location of each spot in the array by assigning the coordinates of the center of each spot. Automating this part permits high-throughput analysis. The next step is segmentation;foreground,the set of pixels corresponding to labeled cDNA hybridizing to its complementary DNA sequence spotted on the microarray, is separated from the background. The last step is quantification. In this step, an intensity value is assigned to each spot by measuring the average intensity of the pixels.

After image conversion, it is recommended to check the quality of the whole array as well as the individual spots within an array. The quality assessment is often carried out in serveral steps. First, visual inspection is done by examining the image plot, in which each image pixel correponds to a spot. If no obvious nonrandom patterns that would suggest poor data quality is observed, the image is passed. Second, numerical methods are used to check whether the spot and background intensities satisfy some quality crieria such as whether the background are uniformly distributed or clustered together or displayed some pattern. Third, to ensure the accuracy and precision of an experimental process is maintained, a quality control procedure should be performed. A simple quality procudere can be done by plotting an image graph to detect specific problems with the array and making a side-by-side display of boxplots of the sequence of arrays to detect specific problems across arrays. Last, the assessment of quality of the indivisual spots can be done by studying the properties of the intensity distributions of spots and if replicates are availble, replicates spots can be analyzed to check whether any value is significantly different from the others.

It is assumed that a spots measured intensity includes a contribution not

specifically due to the hybridization of the target to the probe, for example non-specific hybridization and fluorescence emitted from other chemicals on the glass, called the background fluorescence. Thus we would like to measure and remove such contribution to obtain a more accurate quantification of hybridization which is called background adjustment. Background-adjusted spot intensity values are the results of substracting background from the raw spot intensity values. There are several background estimating methods.

Global background adjustment. In ths approach, the background is estimated as the average intensity of all the pixels not belonging to apots. Its usage is limited because usually the background is not uniform over the entire microarray.

Spot background adjustment. The spot background is substracted from the spot intensity value in this approach. However, the spot and the background are usually imperfectly separated and there exists strong correlation between spot intensity and background intensity. So this approach is rarely effective.

Smoothed background adjustment. Experimental effects, the causes of true variation in background, vary gradually across the slide so the background should be smooth and it could be smoothed by running a simple smoothing procedure. Yang et al.(2000) describe a sophisticated smoothing procedure called mopholog-ical opening.

Once the background intensity of the gth spot BI_g is estimated, the background-adjusted spot intensity value AI_g can be obtained from the equation:

$AI_g = SI_g - BI_g$, where SI_g is the spot intensity at the gth spot.

In some cases, BI_g can exceed SI_g, which causes a negative value for AI_g. One way to avoid it is to make an adjustment. For an instance, if T is a low percentile of the SI_g values, let $AI_g = max(SI_g - BI_g, T)$.

1.2.2 Data Preprocessing

After the images are converted into quantitative data, microarray data still can not used directly for data analysis without data preprocessing due to the immaturity of microarray technology. Data preprocessing addresses three data-related issues: to transform the data into a scale suitable for analysis, to remove the effects of systematic sources of variation and to identify discrepant observations and arrays.

1. Data transformation. The logarithmic transformation,i.e. $X \to log_(X)$, is one of the most widely used transformations. It has the advantages of imprving variance estimation and reducing the skewness of highly skewed distribution. Also it is easy to interpret the log ratios as log fold changes. Speed (2001) recommends the use of logarithmix transformation, but it may not be defined over the full range of data. An alternative approach, $X \to log(X + c)$ where c is a positive constant could be considered. Both transformaton methods are subject to the problem that they can inflate the variance of observations whose means are near 0.

Given the drawbacks of logarithmic transformation, a transformation for microarray data which stablize the variance over the full range of data is suggested. Durbin et al.(2002) propose the following model:

$X = \alpha + \mu e^{\eta} + \epsilon$,

where X is the measured expression for a single observation for a given gene on a array, α is the mean background for the given array and the sample, μ is the true expression level, and η and ϵ represent normally distributed error terms with mean zero and variances σ_{η}^2 and σ_{ϵ}^2 respectively.

At low expression levels, which μ is close to 0, the measeured expression can be written as $X \approx \alpha$ and X is approximately normally distributed with mean α and variance σ_{ϵ}^2. At high expression levels, the measured expression can be written as $X \approx \mu e^{\eta}$ and X is distributed approximately as lognormal with variance $\mu^2 S_{\eta}^2$,

where $S_\eta^2 = e^{\sigma_\eta^2}(e^{\sigma_\eta^2} - 1)$. In a log scale, $log(X) \approx log(\mu) + \eta$, which implies that the variance of log(X) is a constant. At moderate expression levels, the measured expression follows a mixture distribution of a normal and a log-normal. Its variance is $\mu^2 S_\eta^2 + \sigma_\eta^2$ which varies with μ.

2. Data normalization. During the complex and intricate microarray experimental process, there are many systematic effects are introduced into the intensity measurements. Among these effects, the most severe one is caused by different labelling efficiencies across different arrays. For example, suppose there is one control sample and one treatment sample and gene A is not differentially expressed between the two samples. However, because of different labelling efficiencies of the two sample RNAs, the intensities of gene A in the two sample arrays are observed as 10× and 1× respectively, resulting in an apparent differentiation. In order to improve the comparabilities among different microarrays, we need to try ans remove the effects of such systemtic variations and process the data from different microarrays to a common scale. Following the example, if gene A intensities are adjusted for gene B, whose intensities in the two saple arrays are observed as 100× and 10× respectively also due to different labelling efficiencies across the two samples, then this bias is eliminated. This self-control normalization method is widely used in the statistical world of microarray data analysis. Normalization by the sum of intensity is an example, which is to sum the intensities of all spots in an microarrays and to normalize individual spots by that sum. An equivalent idea is normalization by the mean of intensity. Similar, but not equivalent, approaches are normalization by the median, normalization by the log of median, normalization by the third quantiles, normalization by scaled z score and so on. The rationale for these normalization schemes lie in that some quantitiative values should be roughly the same across arrays. Although the assumption may not hold well if there are only a few genes under investigation, it might be very reasonable if the expreesion profiling for up to thousands of genes

are included in the analysis and only a small proportion of them are believed to be differentially expresed. In addition to the above global normalization schemes which are intensity-independent, there are intensity-dependent normalization approaches in which the the data is normalized through a nonlinear normalized function: $X \rightarrow f(X)$. The representives of intensity-dependent normalization are smooth function normalization and quantile normalization. In the former approach, a smooth function h_i, inverse of f_i is estimated by fitting the model

$X_{gi} = h_i(M_g) + \epsilon_{gi}$ to invariant gene sets, where X_{gi} is the transformed spot intensity for the gth gene in the ith array, $M_g = median(X_{g1}, ..., XgI)$ and ϵ_{gi} is random error. Then the normalized value can be obtained form $X'_{gi} = f_i(X_{gi})$.

Different choices of smooth functions are cubic splines, lowess smooth functions etc.

In quantile normalization, the distributions of the transformed spot intensities of all micrarrays are forced to be equal. Bolstad et al(2003) propose an algorithm to equate quantiles which is including three steps: (1) sort intensities in each array; (2) compute mean intensity t each rank across the arrays; (3) Replace each intensity by the mean intensity at its rank. They also show that quantile normaliation perfoms best and the lowess normalization is comparable to quantle normalization.

3. Outlier identification. In microarray data, an outlier is an observation that is markedly different from the majority of the other values for that gene. Due to the problem of high dimensional multicariate measures in each sample microarray but only a few samples available, detection of outliers is a challenging job.

Because they are themselves influenced by outliers, classical tools based on the mean and standard deviation,for example z-score, are rarely able to detect outliers (masking effect) and are possible to misclassify normal points as outliers (swamping effect). More reliable approaches are based on median and the MAD(medan absolute deviation) which are resistant to outliers, for an instance;

The resistant z-score rule. Calculate a resistant z-score z^*_{gi} for every observation:

$z^*_{gi} = \frac{X_{gi}-\tilde{X}_g}{\tilde{s}_g}$, where $\tilde{X}_g$ and $\tilde{s}_g$ are the median and MAD of the gth gene. Call X_{gi} an outlier if $|z^*_{gi}|$ is large. With very few replicates, the MAD is not dependable estimate of the scale of the data. As an adjustment, we calculate a smoothed vresion of MAD, $\tilde{MAD}_g$, in the following way. First the absolute divations from the median:$AD_{gi} = |X_{gi} - \tilde{X}_g|$, then run a smoother through the relationship of AD_{gi} versus $\tilde{X}_g$ and the fitted value, $\tilde{MAD}_g$, is used as an estimator of scale for the gth gene. That is the revised rule:

The revised z-score rule. Calculate a revised z-score z^{**}_{gi} for every observation: $z^{**}_{gi} = \frac{X_{gi}-\tilde{X}_g}{\tilde{s}'_g}$. Call X_{gi} an outlier if $|z^{**}_{gi}|$ is large.

1.2.3 Data Analysis

Once the scanned image has been processed and the initial data has been transformed, normalized and consistantly checked, formal statistical anlysis of data could be done to extract information for certain use. Application of statistical methodology is feasible when the microrray experiments can be performed on replicate samples. Unfortunately, while there are tens of thousands of genes in data, the replicate is small so that the information content per gene is small. The typical characteristc of microarray data that the number of genes measured is much greater than the avaliable sample size imposes a serious challenge to statisticians. Either standard statistical tools need to be tailored or extended to tackle microarray data or new approaches have to been developed specially to handle such data.

The rest of this section is focused on reviews of some popular statistical methods published in the literature. Since the research in microarray data analysis has grown dramatically during the past serveral years and is related to many other disciplines such as genomics, bioinformatics and statistical experimental designs,

it is not realistic to have an exhaustive review of all of these areas. Instead, only those methods with substantial statistical components from the prospective of genetic statisticians are reviewed here.

Gene-wise Comparisons. Gene-wise Comparisons across two or more conditions is a major and popular statistical analytic task in microarray data analysis, in particular, to identify significantly differentially expressed genes across these conditions. As an example, one experiment might be conducted to identify which genes are differentialy expressed in diseased cells versus normal cells, which would enable biologists to identify genes associated with the disease process and enable the development of drugs targeted to the difference between diseased and normal cells. We first consider the simplest and most common case: a comparison between the gene expression profiles of two groups. And we assume that the data is suitably transformed and normalized. The notation used are as follows: Let X_{gij} denote the intensity measurement for the g^{th} gene in the i^{th} microarray in the j^{th} group, where i=1,...,n_j; j=1,2; and g=1,...,G. Moreover, let $\bar{X}_{gj}$, $\tilde{X}_{gj}$, $\bar{s}_{gj}$ and $\tilde{s}_{gj}$ denote the mean, median, standard deviation, and the median absolute deviation (MAD) from the median of gene g in the j^{th} group respectively.

Two Sample T-test. The two sample t test is the most basic statistical gene-by-gene comparison approach.

For gene g, the two sample t test statistic to test

$H_{0g} : \mu_{g1} = \mu_{g2}$ vs $H_{1g} : \mu_{g1} \neq \mu_{g2}$

is given by $T_g = \frac{\left|\bar{X}_{g1} - \bar{X}_{g2}\right|}{S_g \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$

where $S_g^2 = \frac{(n_1-1)s_1^2 + (n_2-1)s_2^2}{(n_1+n_2-2)}$ is the pooled estimate of variance.

Under the assumption that $X_{g11}, X_{g21}, \ldots, X_{gn_11} \sim N(\mu_{g1}, \sigma_g^2)$ and

$X_{g12}, X_{g22}, \ldots, X_{gn_22} \sim N(\mu_{g2}, \sigma_g^2)$, the null distribution of T_g is a t distribution with degrees of freedom $\nu = n_1 + n_2 - 2$. The p-value is calculated by $p_g = Prob(|T_g| > T_{g,obs})$, where $T_{g,obs}$ is the observed value. A gene is declared to

be significantly differentially expressed at a specified level α if $p_g < \alpha$.

Genes express different levels and the variablity of a gene may depend on its expression level which means that the equal variance assumption is likely to unhold. To overcome the this problem, the unequal-variance version of two sample t test, called Welch's test, is proposed. The statistic of Welch's test is:

$T_g^* = \left|\bar{X}_{g1} - \bar{X}_{g2}\right| / \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}$,

and the null distribution of T_g^* is approximately a t distribution with degrees of freedom:

$v = \left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2 / \left[\frac{1}{(n_1-1)}\left(\frac{s_1^2}{n_1}\right)^2 + \frac{1}{(n_2-1)}\left(\frac{s_2^2}{n_2}\right)^2\right]$,

The p-value is calculated by $p_g^* = Prob(|T_g^*| > T_{g,obs}^*)$, where $T_{g,obs}^*$ is the observed value. A gene is declared to be significantly differentially expressed at a specified level α if $p_g^* < \alpha$.

Welch's test is less powerful than its equal-variance version because it has fewer degrees of freedom. A drawback of both method is that the dependence between the t test statistic and the standard error estimate leads to an hight false positive rate for low variance genes and an high false negative rate for high variance genes given that the sample size per group is very small in typical gene expression data. Also how to estimate standard errors well is another issue issue.

Statistical Significance of Microarray(SAM) Tusher et al.(2001) suggested an approach to overcome the drawbacks of standard t test especially the last one shown above. They added a small positive constant, called fudge factor to the denominator of the t statistic. The adjusted t statistic of the gth gene is

$T_g = |\bar{x}_{g1} - \bar{x}_{g2}| / (s_g + s_0)$,

where s_g is the pooled standard error and s_0 is the fudge factor. The value for s_0 is chosen to minimize the coeffiecient of variation of T_g, which is computed as a function of s_g. The procedure is as follows: Let s^α be the α percentile of the s_g values and let $T_g^\alpha = r_g/(s_g + s^\alpha)$. Compute the 100 quantiles of the s_g values, denoted by $q_1 < q_2 ... < q_{100}$. For $\alpha \in (0, 0.05, 0.1, ...1.0)$, compute the mad

(median absolute deviation from the median), $v_j(\alpha)$, of the T_g^α values within the interval $[q_i, q_{i+1}]$ for i = 1, 2, ..., n. Then compute the coefficient of variation of the $v_j(\alpha)$, $cv(\alpha)$. Choose $\hat{\alpha}$ is the one that minimize $cv(\alpha)$. And now $\hat{s}_0$ is fixed as $s^{\hat{\alpha}}$.

Once the fudge factor is determined, adjusted expected t statistics are computed by taking the means of multiple sets of adjusted permuted t statistics, which are calculated using the same formula as described above except that permuted data are used for these calculations. Creating expected data under null distribution through permutation is a widely adopted technique in microarray data analysis. A version of permutated data is produced by permuting the response data. In the other word, it assigns the same number of observed cases to the study population and the rest are viewed as controls. This resampling procedure is repeated many times. Then the adjusted observed and expected t statistics are ordered respectively and the difference between the two at each ordered location is calculated. If the difference is larger than the fixed cut-off value, the corresponding gene is called "significant positive" and if the difference is negative and its absolute value exceeds the threshold, the corresponding gene is called "'significant negative".

Next the total number of significant genes and the median, k_m, (or the 90th percentile, $k_{0.9}$) number of falsely called genes are computed and the proportion of true null genes in the data set π_0 is estimated. The positive false discovery rate (pFDR) is calculated by the multiple of k_m (or $k_{0.9}$) and π_0 divided by the number of significant genes.

Despite the advantage mentioned above, the performance of this approach depends on the assumption that the expected and observed orders of the majority of those non-differentially expressed genes are the same, while the small proportion of differentially expressed genes are all located at the extremes. This assumption is hard to hold without careful experimental design.

	declared significant	*declared non-significant*	*Total*
null	U	S	G_0
Alternative	V	M	G_1
Total	R	A	G

Table 1.1: Possible Outcomes of Testing

Multiple Testing Adjustment. Since a statistical test is being run on every gene, doing gene-wise comparisons involves performing a very large number of tests simutaneously. One drawback of conducting so many tests is that the more the number of statistical tests perfprmed, the higher the expected number of false positives. Without making a suitable multiple testing adjustment, the number of false positives can be high enough to overhelm the actual effects. So we need to adjust the individual p-values of the tests to possibly alleviate this probelm. We will outline several ways of adjusting the p-values for the increased false positive rate do to multiple testing.

Consider the situation that G statistical tests have been performed, possible outcomes are shown in Table1.

If no adjustment is made for multiple testing, we could control the per-comparison error rate (PCER): **PCER** = E(V)/G. A common multiplicity adjustment attempts to control the familywise error rate (**FWER**). The FWER is defined to be the probability of making at least one false positive error: FWER = $Prob(V \geq 1)$. Rejecting each individual test with a type I error rate of α/m guarantees, by Bonferronis type of argument, that FWER is controlled at level α in the strong sense, i.e. $FWER \leq \alpha$ for any combinations of null and alternative hypotheses. Benjamini and Hochberg (1995) proposed another type of error to control **FDR**, which is defined to be the expected proportion of false positives among the rejected hypotheses:

$$\text{FDR=E[Q] and } Q = \begin{cases} V/R, & R > 0 \\ 0, & R = 0 \end{cases}$$

factors	*change*	*sample size change*
Variability of Population	↗	↗
Desired Detectable Fold Changes	↗	↘
Power	↗	↗
Error Rate	↗	↘

Table 1.2: Relationship between Sample Size and Four Factors

Storey (2002) proposed to control positive FDR (**pFDR**), i.e.

pFDR= $E\left(\frac{V}{R}|R>0\right) = \frac{FDR}{Prob(R>0)}$

The pFDR states the fact that an adjustment is necessary when there are positive findings.

Sample Size Calculation. When planning microarray experiments, an often-asked question is how many samples required to ensure adequate statistical power. This applied topic attracts many statisticians. Generally speaking, there are four factors that affect sample size: 1)Variability of Population; 2)Desired Detectable Fold Changes; 3)Power to Detect the Differences; 4)Error Rates. The relationship between sample size and the four factors are shown in Table1.2

The general procedures are to select a gene selection method (t-test, mixture model, SAM etc.) and to select one or more reliability criteria (FDR, FWER, Sensitivity etc.). Serveral articles have addressed the sample size problem.

Pan et al (2002) proposed a mixture model approach. Unfortunately, there is a major flaw in their approach.

Their model: $X_{ji} = \mu_{1i} + \epsilon_{ji}, Y_{li} = \mu_{2i} + e_{li}$, where X_{ji} is the j^{th} sample expression of the gene i in first group and Y_{li} is the l^{th} sample expression of the gene i in second group.

And $E(\epsilon_{ji}) = E(e_{li}) = 0, Var(\epsilon_{ji}) = \sigma_{1i}^2, Var(e_{li}) = \sigma_{2i}^2$.

Null Hypothesis: $H_0 : \mu_{1i} = \mu_{2i}$

Test Statistic: $Z_i = \frac{\sum_{j=1}^{G} X_{ji}/G}{\sigma_{1i}} - \frac{\sum_{j=1}^{G} Y_{ji}/G}{\sigma_{2i}} = \frac{\mu_{1i}}{\sigma_{1i}} - \frac{\mu_{2i}}{\sigma_{2i}} + \frac{\sum_{j=1}^{G} \epsilon_{ji}}{G\sigma_{1i}} - \frac{\sum_{j=1}^{G} e_{ji}}{G\sigma_{2i}}$

$E(Z_i) = \frac{\mu_{1i}}{\sigma_{1i}} - \frac{\mu_{2i}}{\sigma_{2i}}, Var(Z_i) = \frac{2}{G}$

The test statistic can not be used to test the null hypothesis unless $\sigma_{1i} = \sigma_{2i}$.

Zien et al (2003) constructed a model including several different sources of error. The model itself is non-identifiable, because the authors suggest heuristic choices for the main parameters in the model insted of using historical data to directly estimate the necessary parameters. Since the behavior of microarray data from different microarray technologies and for different biological systems is extremely varied, it is not reasonable that considering the technical errors (measurement errors, i.e., the technical part of the variability of measurement results) and biological variability (biological variation of individual genes, both between and within classes of samples) of all microarrays are following same models.

Tsai(2005) estimated sample size using two sample z-test and control the sensitivity, true discovery or accuracy. But the sample sizes estimated is very samll and is much less than that needed in practice when controlling true discovery or accuracy. For example, if V=1, G=10000, $\pi_0 = 0.95$, power $\phi = 0.80$, true discovery rate=95%, the required sample size was 3. Let's see why the sample size was small. True discovery rate= U/R = (R-V)/R = (R-1)/R. So if we can identify 20 differentially expressed genes, the true discovery rate would be 19/20=95%. Of course, the sample size required to identify 20 out of 500 differentially expressed genes would be small. Another example, if V=1, G=10000, $\pi_0 = 0.95$, power $\phi = 0.80$, accuracy=95%, the required sample size was 1. The sample size was too small because accuracy= $(G_0 - V + U)/G = (0.95 * 10000 - 1 + U)/G > 0.95, if U > 1$. The sample size required to identify at lease 1 out of 500 differentially expressed genes would be very small. These two examples show that neither true discovery nor accuarcy is a good reliability criteria. sensitivity, defined as the fraction of truly differentially expressed genes identified at the desired power, is much better according to their sample size table. For an instance, if V=1, G=10000, $\pi_0 = 0.95$,

power $\phi = 0.80$, sensitivity=U/G=80%, sample size was 12. The means that, if we want to identify 80% of the differentially expressed with 80% power, the sample size required was 12.

An intersting resampling method is proposed by Shuying S.Li et al (2005). The authors calculated power of the FDR-controlling procedures when array datasets with a pre-specified size were generated by reampling method, which maintained the correlation among genes. They compared the results of two direct FDR-controlling procedures (BH and ST) and four reampling-based FDR-controlling procedures (BH, ST and other two appraches discussed in Yekutieli and Benjamini(1999)). The simulation results showed that the actual FDR could be two times higher than the pre-specified level if $(1-\pi_0)$ is small (<1 percent) and gradually achieves the pre-specified level as $(1-\pi_0)$ increases to 20 percent. In either the direct or the resmapling-based appraoch, π_0 is under estimated except when $\pi_0 = 0.80$. Since these testing procedures under-control the actual FDR when the genes are correlated and the proportion of differenctially expressed genes is less than 20 percent, in practice, they recommended adjusting the pre-specified level. For example, it is recommended to adjust the pre-specified level by half if the proportion of positive genes is below 10 percent.

Shao and tseng(2007) presented a method that make dependence adjustment to one-sided z-test controlling FDR. It assumes many small dependent blocks in the arrays. The advantages of this appraoch are correlation among DE genes considered and FDR level controlled. The BH procedure controls the FDR at the nominal level and the ST procedure yields more liberal FDR. One drawback is that we need the correlation among test statistics of differentially expressed genes as input. In general, we know or we can estimate the proportion of differentially expressed genes from pilot data, but we do not know which genes are differentially expressed so that we. Therefore, this approach is not feasible in explorotary analysis in which we can not provide the correlation matrix of differentially genes

which is needed by the procedure.

Two more papers discussing sample size calculation controlling the FDR are Pawitan et al (2005) and Liu and Hwang (2007).

Cluster Analysis is a useful multivariate data analysis techniques for the analysis of microarray data. It organizes the entirety of genes into an assortment of clusters so that the genes that behaved the most similarly in the experiment will be members of the same cluster, while genes that behaved differently will be members of different clusters. The principle is that it is reasonable to expect that genes performing similar functions or operating in the same genetic pathway would behave similarly across conditions. Since the seminal paper by Eisen et al (1998), a wide range of clustering approaches have been developed in the context of microarray data such as hierarchical clustering, partitioning methods, and model-based clustering etc.

Hierarchical Clustering is the most popular clustering method. It produce a hierarchy of clusters, called a tree or dendrogram, in two dstinct ways: bottom up (agglomerative clustering) and top down (divisive clustering). Agglomerative algorithm begins with each gene in its own cluster, then the closest pair of clusters are combined whereas d ivisive algorithm begins with all genes situated in one giant cluster, then the loosest cluster is divided into two clusters. Although in principle, bottom-up clustering process can be ontinued until all genes are in one cluster and top-down clustering process can be continued until each gene in its own cluster, typical clustering process ends when it reaches the desired clusters or a specific criteria is satisfied. Various hierarchical clustering methods with their application to microarray data have been discussed by Eisen et al (1998), Cheng and Church (2000), Friedman and Meulman (2002), Madeira and Oliveira(2004), Chipman and Tibshirani(2006), Higham et al.(2007) and Nowak and Tibshirani (2008).

Partitioning method splits the genes into a specified number of mutually exclusive and exhaustive groups. It iteratively reallocate the observations to clusters until some criterion is met, for example, minimize within-cluster sum of squared dissimilarities. Examples of application of partitioning methods to microarray data are: Tamayo et al (1999) and Toronen et al (1999) used self-organizing map (SOM) for clustering microarray data; Dudoit and Fridyland (2002) applied k-medoids to gene expression data; Dembele and Kastner (2003) and Asyali and Alci(2003) cluster microrray data by fuzzy c-means. Tseng (2007) used penalized and weighted k-means in clusering gene expression data.

Model-based clustering has been applied to microarray data by Yeung et al.(2001), McLachlan et al.(2002), Pan et al.(2002), Medvedovic and Sivaganesam (2002), Medvedovic et al.(2004) and Pan (2006). It is a partitioning method which assumes that each cluster is generated by a probability distribution such as multivariate normal, mixed normal, gamma etc.. The advantage of model-based clustering is that we do not need to heuristically judge which clustering result is the best which has to be done with most other clustering procedures. One could fit the model with different parameter values of probability density functions and then pick up a best model according to a specific criterion function such as AIC and BIC.

Gene Co-expression Netwoek. Graph-theoretic approaches are increasingly used to explore the functionality of genes. An example is gene co-expression network which tends to exhibit modular structure grouping together genes responsible for individual biological processes and functions. In this sense, gene co-expression network provides the interaction between individual genes and a system-level view of the organism.

The concept of gene co-expression network is quite straightforward: nodes represnt genes and nodes are conneted if the correponding gene pairs are significantly co-expressed. Generally, the connections between genes are converted from

a co-expression measure, for example, the absolute value of Pearson correaltion coefficient which is the most commonly used co-expression measure. Serveral researchers have suggested to threshold this Pearson correlation coeffficient to construct gene co-expression networks (Butte and Kohane,2000; Carter et al 2004; Davidson et al.,2001). There are two ways to pick a threshold: one way is picking a 'hard' threshold (a number) based on the notion of statistical significane so gene co-expression is encoded using binary infromation(conneted=1, unconnected=0);the other way is called 'soft' thresholding which weighs each connetion by a number between 0 and 1. The drawbacks of 'hard' thresholding include loss of information of the magnitude of gene connections and sensitivity to the choice of the threshold(Carter et al.,2004). Moreover, an important question is whether it is biologically meaningful.

'Hard' threshodling results in unweighted networks while 'soft' thresholding results in weighted networks. After thresholding, connection strength among genes are produced. The resulting matrix is used to define a measure of node dissimilarity (distance). The node dissimilarity measure will be used as input of a clustering method to define network modules(clusters of nodes). Once the modules have been defined, one can build a gene network and define additional network concepts. Finally, the modules and their highly connected genes are often related to external gene information. For example, the highly correlated genes of a certain module could predict cancer survival (Mischel et al.,2005).

1.3 Summary and Commonly Used R Packages

Today, microarrays are manufactured by multiple companies (two of which are Affimetrix, and Agilent Technologies). Related analyses of the information gained from microarrays help provide answers to many questions such as:

* How does diseased tissue differ from normal tissue?

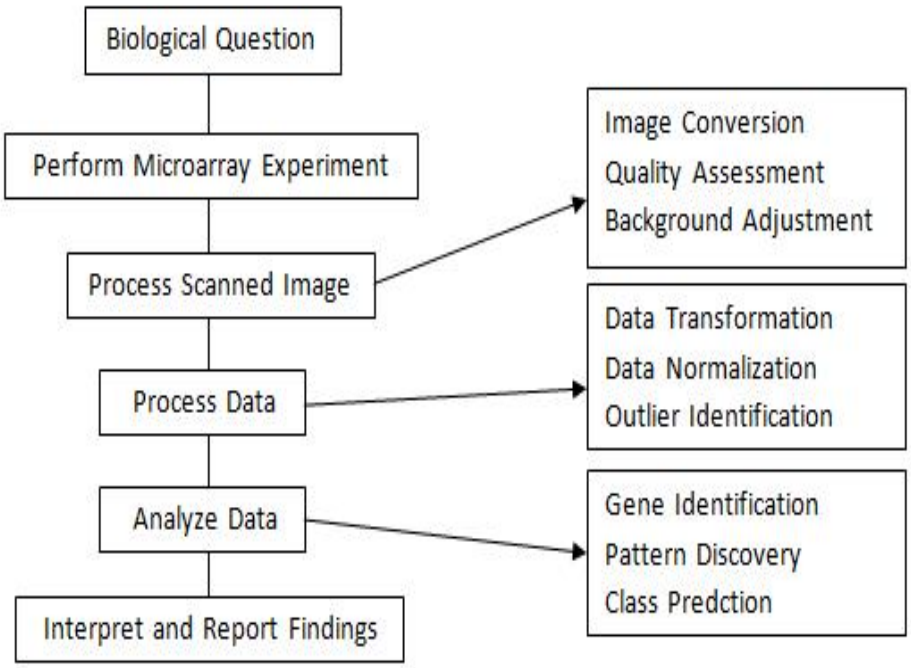

Figure 1.1: Schematic of a typical microarray data analysis

* What stage of a disease is present in a person?

* Which drugs will be best for treatment?

* How can drugs be improved for treatments?

* Which drugs work best at different stages of a disease?

* Which genes are acting together, as a cluster?

* What or which groups of genes are responsible for a hereditary characteristic, hereditary syndrome, or disease?

Figure 2.1 shows the schematic of a typical microarray data analysis. This thesis focus on the statistical methods for the Analyzing Data step. R is the most commonly used softwares and it contains a number of packages to analyze microarray data. Table 1.3 lists a few of them.

Table 1.3: Most Commonly Used R Packages

USAGE	*STATISTICAL METHODS*	*R PACKAGES*
Gene Identification	(Improved) Conditional T test	DNAMR
	SAM	samr
Pattern Discovery	Model-based clustering	mclust
Class Prediction	Random Forest	RandomForest
	Neural Network	nnet
	Support Vector Machine	e1071

Chapter 2
Stochastic Approximation for Distributions

Abstract Stochastic approximation is proposed by Robbins and Monro (1951) to find the solution to $M(x) = \theta$ given θ is known and $M(x)$ is an unknown monotone function. It finds a series of $x_1, x_2, \ldots$, in such a way that x_n, will tend to x^* in probability. It is very powerful in most cases. however, it does not work for distribution because its intermediate estimator may not be a distribution. Thus target estimation is induced to solve this roblem. The newly proposed method is called S.A.D.. We present good propetties of S.A.D. and give some examples.

2.1 Introduction

Stochastic approximation is proposed by Robbins and Monro (1951). Let $M(x)$ denote the expected value at level x of the response to a certain experiment. $M(x)$ is an unknown monotone function of x and a solution $x = \theta$ to $M(x) = \alpha$ is needed. Stochastic approximation method gives successive $x_1, x_2, \ldots$, in such a way that x_n, will tend to θ in probability.

Suppose for each value x, $H(y|x)$ is a distribution function in y such that $M(x) = \int_{-\infty}^{\infty} y dH(y|x)$ and there exists a positive constant C such that $Pr[|Y(x)|] = \int_{-C}^{C} dH(y|x) = 1$ for all x. A nonstationary Markov chain x_n is defined as follows: x_1 is an arbitrary constant; $x_{n+1} - x_n = a_n(\alpha - y_n)$, where y_n is a random variable such that $Pr[y_n|x_n] = H(y|x_n)$. Let $b_n = E(x_n - \theta)^2$ and $\lim_{n\to\infty} b_n = b$.

Theorem 2 in Robbins and Monro (1951) tells us if a_n is of type $1/n$ and if M(x) satisfies

(1) M(x) is nondecreasing (2) $M(\theta) = \alpha$ (3) $M'(\theta) > 0$ then $b = 0$.

2.2 Stochastic Approximation for Distributions

Now if M(.) is a funtional of continous distributions, we need to find a solution $F^{\theta}(x)$ such that $M(F^{\theta}(x)) = F^{\alpha}(x)$ where $F^{\alpha}(x)$ is a given distribution function. A straghtforward way to solve this problem using stochastic approximation is to calculate

$F^{n+1}(x) - F^{n}(x) = a_n(F^{0}(x) - M(F^{n}(x)))$

and get $F^1(x), F^2(x), \ldots, F^n(x), \ldots$ such that $F^n(x)$ converges given that some specific conditions hold. But there is a problem here since F_{n+1}, the intermediate estimator may not be a distribution function. A simple example is that F_{n+1} could be bigger than one or less than zero.

Think the other way, if $M(F^{\theta}(x)) = F^{\alpha}(x)$ can be described as

$h(q_{(i/m)}) = s_{(i/m)}$ $i = 1, 2, \ldots, m$,

where $h : [-1, 1] \to [-1, 1]$ is a monotone increasing function,

$q_{(i/m)}$ is the $(i/m)^{th}$ quantile of $F^{\theta}(x)$ and $s_{(i/m)}$ is the $(i/m)^{th}$ quantile of $F^{\alpha}(x)$.

then $q_{(i/m)}$ can be approximated by $q^{n+1}_{(i/m)} = q^{n}_{(i/m)} + a_n(q^{0}_{(i/m)} - E(q^{n}_{(i/m)}))$ when specific conditions are satisfied.

This stochastic approaximation method used to estimate distribution functions is called S.A.D.

2.3 An Example

2.3.1 Original Problem and Algorithms

This problem comes from microarray data analysis. Microarray model: $X_{gij} = \mu_{gj} + \sigma_g \epsilon_{gij}$, where X_{gij} is log transformed and suitably normalized intensities,

μ_{gj} is the mean of the gth gene in the jth group, and σ_g^2 is the variance of the gth gene. Also, g(g=1...G) indexes the genes on the microarray, j(j=1,2) indexed the groups, and i (i=1...n_j) indexes the objects. We want to estimate F_σ, where F_σ is the distribution of σ_g and F_σ is the same for all the groups and all the genes. Although one can use the emprical distribution of σ_g as the distribution estimator, it is not accurate because the estimate of σ_g for all g comes from a sample standard deviation of few samples.

In Amaratunga and Cabrerra(2003)(2007) and Cabrera and Yu(2007), there is an old algorithm:

a) Generate a null distribution for the data by substracting the sample means and dividing by the standard deviations.

b) Calculate the emprical distribution of s_g, say $\hat{F}_s$. Assume that $\hat{F}_s(x)$ is the true distribution of σ, then resample from the null distribtuion of x and multiply each sample by a σ generated from $\hat{F}_s(x)$. Repeat this 10,000 times and get 10,000 pairs of samples.

c) From each pair of samples, calculate a value for the pooled sample standard deviation, namely s_g^*, for $g = 1, \ldots, 10,000$. Let $\hat{F}_{s^*}(x)$ be the empirical distribution of the $s_g^{*'}s$. Then the estimator of g is obtained by mapping the emprical distribution $\hat{F}_s(x)$ into $\hat{F}_{s^*}(x)$. More in detail, $\hat{g}(y = \hat{F}_s(x)) = \hat{F}_{s^*}(\hat{F}_s^{-1}(y))$ and $\hat{g}^{-1}(y) = \hat{F}_s(\hat{F}_{s^*}^{-1}(y))$.

Therefore, the estimator of F_σ is $F_\sigma(x) = \hat{F}_s(\hat{F}_{s^*}^{-1}(\hat{F}_s(x)))$.

This old algorithm does not work very well in some cases, so we developed the following new algorithm:

(1) For $g = 1, ..., G$, calculate the sample deviation $\hat{s}_g^{(0)}$ of

$(x_{g11}, \ldots, x_{g1n_1}, x_{g21}, \ldots, x_{g2n_2})$.

Let $\hat{s}^{(0)} = (\hat{s}_1^{(0)}, \hat{s}_2^{(0)}, \ldots, \hat{s}_G^{(0)})$.

The empirical distribution of $\hat{s}_g^{(0)}$, say $\hat{F}^{(0)}$, will serve as the initial estimator of F_σ.

(*) For k, sample $\sigma_g^{(k-1)}$ from $\hat{F}^{(k-1)}$ and sample $x_{g11}^{(k-1)}, \ldots, x_{g1n_1}^{(k-1)}, x_{g21}^{(k-1)}, \ldots, x_{g2n_2}^{(k-1)}$ independently from $N(0, \sigma_g^{2(k-1)})$.

Calculate sample devation $\hat{s}_g^{(k-1)}$ of $x_{(g11}^{(k-1)}, \ldots, x_{(g1n_1}^{(k-1)}, x_{(g21}^{(k-1)}, \ldots, x_{(g2n_2}^{(k-1)}$ and empirical distribution of $(\hat{s}_1^{(k)}, \hat{s}_2^{(k)}, \ldots, \hat{s}_G^{(k)})$.

Repeat a few times and get $E(\hat{F}^{(k-1)})$

(2) For $k = 1, 2, \ldots, K$ (a spcified number),

Let $\hat{F}^{(k)} = \hat{F}^{(k-1)} - \frac{1}{n+1}(E(\hat{F}^{(k-1)}) - \hat{F}^{(0)})$, where $E(\hat{F}^{(k-1)})$ is estimated by (*).

Then $\hat{F}^{(K)}$ is our final estimator of F_σ.

2.3.2 Simulation

Without loss of generality, we simulate $\sigma_g^2 \sim \chi_{10}^2/10$ and $x_{g1}, x_{g2}, \ldots, x_{g6} \sim N(0, \sigma_g^2)$ iid, for g=1,2,...,10000. It is found that as the interation goes, our estimator goes closer and closer to true distribution. And it converges pretty fast. Figure 2.1 shows the results, where the green line is the true distribution function, the black one is the empirical distribution of $\hat{s}_g$ and the red one is our estimator after 8 interations.

2.4 Extention to Correlation Matrix Estimation

2.4.1 The Origin of the Problem

Correlation matrix estimation is one of the most essential problems in multivariate data analysis. In particular, correlation estimation among genes plays an important role in microarra data analysis. For example, induce correlation correction to gene expression identification could improve the statistical power for detection of differentially expressed genes. Intuitively, more power would be gained among highly correlated genes than independent genes given all other conditions

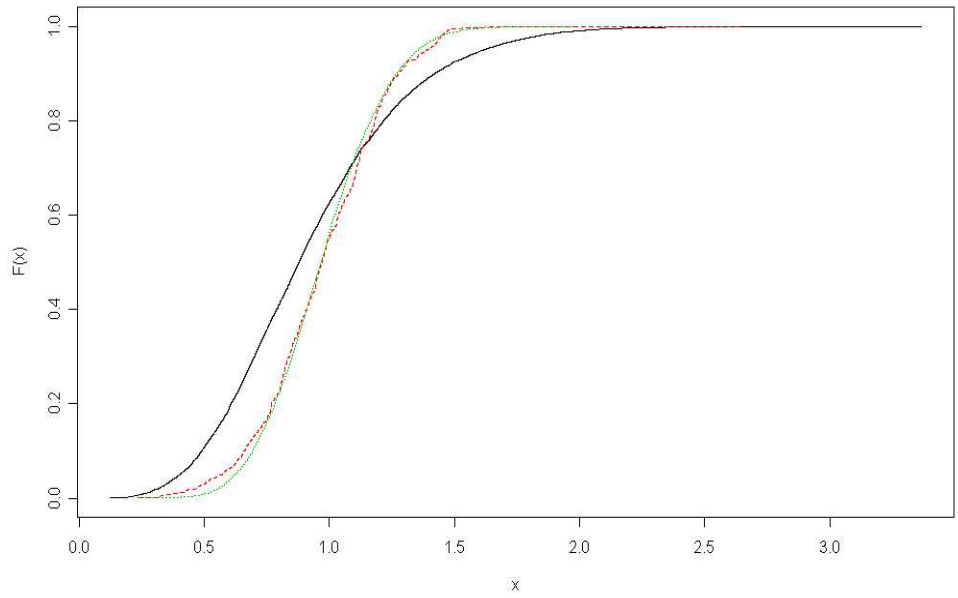

Figure 2.1: Estimators of Distribution. Green:True distribution function; Black: Empirical distribution; Red: Estimator by S.A.D.

are same. Another example, correlation among genes measures the connection strength of gene pairs which leads us to define an unweighted or weighted gene co-expression network, a powerful tool to explore the system level functionality of genes.

Pearson's correlation coefficient is the most popular correlation estimator used in multivariate analysis. However, its performance is poor when applied to microarray data because typical microarray data has thousands of genes with few samples. So we extend S.A.D to estimate correlation matrix of genes in microarray.

2.4.2 Methodology and Algorithm

Target Estimation

Target estimation is explored by Cabrera and Meer(1996), Cabrera and Watson(1997) and Cabrera and Fernholz (1999). The aim of this procedure is to reduce bias and variance of a one dimensional statistic. Cabrera and Fernholz (2003) extended this method to multivariate situations. For a higher dimensional statistic, conditions are given to ensure no bias and lower variance after targeting. We apply a version of this approach for estimating the correlation matrix of genes.

The idea is to estimate the function $h : [-1, 1]^n \to [-1, 1]^n$ definded by $h(\underline{\epsilon}) = \hat{\underline{\epsilon}}$,

where $\Sigma = (\epsilon_{ij})$ is the true correlation matrix of all genes.

$\underline{\epsilon}$ is the vector of lower triangle matrix of Σ.

$\hat{\Sigma} = (\hat{\epsilon}_{ij})$ is the Peason correlation coefficient of all genes.

$\hat{\underline{\epsilon}}$ is the vector of lower triangle matrix of $\hat{\Sigma}$.

Once we have h, $\underline{\epsilon}$ can be estimated by $\tilde{\underline{\epsilon}} = h^{-1}(\hat{\underline{\epsilon}})$ thus we have Σ.

A possible way to estimate h is sampling gene sets, say Y, from multivariate normal distribution with mean 0 and variance $\hat{\Sigma}$. And then calculate the Pearson's correlation coefficient of Y, say $\hat{\Sigma^*}$. After these, h can be got by mapping $\hat{\Sigma}$ into $\hat{\Sigma^*}$. It performs well when the total number of genes is relatively small but it is not so good when there are thousands of genes in the dataset. So we use S.A.D. to estimate function h.

To estimate h, we extend S.A.D. to fit a more complex situation, let y be ordered $\hat{\underline{\epsilon}}$ and x be ordered $\underline{\epsilon}$. The algorithm is as follows:

(A1) Calculate the Pearson Correlation Coefficients $\hat{\Sigma}$ and the vector of lower triangle $\hat{\underline{\epsilon}}$.

(A2) For n=100,101,...,nn(a specified number), compute

$\hat{\underline{\epsilon}}_{n+1} = \underline{\epsilon}_n - \frac{1}{n+1}(E[\underline{\epsilon}_n] - \hat{\underline{\epsilon}})$, $\underline{\epsilon}_{n+1} = sort(\hat{\underline{\epsilon}}_{n+1})$

Calculation of $E[\underline{\epsilon}_n]$:

Suppose we have G genes and m samples, get 100 samples $Z \sim MVN(m, mu = \underline{0}, \hat{\Sigma}_n)$, where $\underline{\epsilon}_n$ is the vector of lower triangle of $\hat{\Sigma}_n$. $E[\underline{\epsilon}_n]$ is the average of ordered vector of lower triangle of cor(Z).

2.4.3 A Simulated Example

We use a very similar simulated example in Zhang and Horvath (2005). The correlation matrix is a block diagonal one:

$$C = \begin{pmatrix} \mathrm{C}_1 & 0 & 0 & 0 \\ 0 & \mathrm{C}_2 & 0 & 0 \\ 0 & 0 & \mathrm{C}_3 & 0 \\ 0 & 0 & 0 & \mathrm{C}_4 \end{pmatrix}$$

where C_m is $n_m \times n_m$ matrix and $n_1 = 100, n_2 = 200, n_3 = 300, n_4 = 500, n = \sum n_i = 1100$.

$C_4 = \boldsymbol{I}$, Identity matrix.

For m = 1,2 and 3,

$$C_m^*(i,j) = \begin{cases} (1 - \frac{0.3 \times max(i,j)}{n_m})^5, & \text{i}\leq 0.95n_m \text{ and } j \leq 0.95n_m \\ (0.85 + \frac{0.3 \times i}{n})^5 \times (0.85 + \frac{0.3 \times i}{n})^5, & \text{i}>0.95\text{n}_m \text{ and } j \leq 0.95n_m \\ (0.85 + \frac{0.3 \times i}{n})^5 \times (0.85 + \frac{0.3 \times i}{n})^5, & \text{i}\leq 0.95n_m \text{ and } j > 0.95n_m \\ 0.95^5, & \text{i}>0.95\text{n}_m \text{ and } j > 0.95n_m \end{cases}.$$

C_m^* is not positive semi-definite, so we keep the eigenvectors and change the negative eigenvalues to a small postive constant and then we get a positive semi-definite matrix. Transform the positive semi-definite matrix to a correlation matrix, that is C_m.

A color coded picture of matrix C is shown in Figure 2.

Our simulated data contains 50 samples which come from multivariate normal distribution, i.e.

$Data \sim MVN(50, \underline{0}, \Sigma = C)$.

First we use target estimation and stochastic approximation to correct the

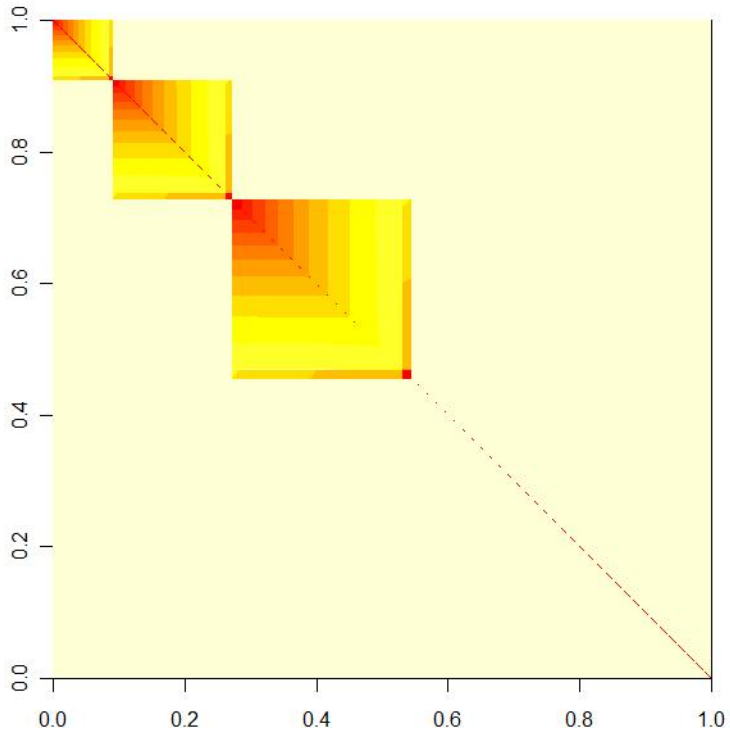

Figure 2.2: Plots of True Correlation

Pearson correlation coefficients of the simulated data. Figure 3 displayed the color pictures of the Pearson correlation coefficients matrix (a) and the corrected one (b) (after 100 interations). It is easily seen that the bias of Pearson correlation coefficients is greatly reduced after correction.

The mean absolute difference between estimated correlation matrix and true correlation is shown in Table 1. From it, it is easily seen that the difference becomes smaller and smaller as the interations are increasing. The mean value of absolute difference between Pearson's correlation coefficents and true correlations is 0.12 at the beginning. While after 161 interations, the mean absolute difference decreased to around 0.05.

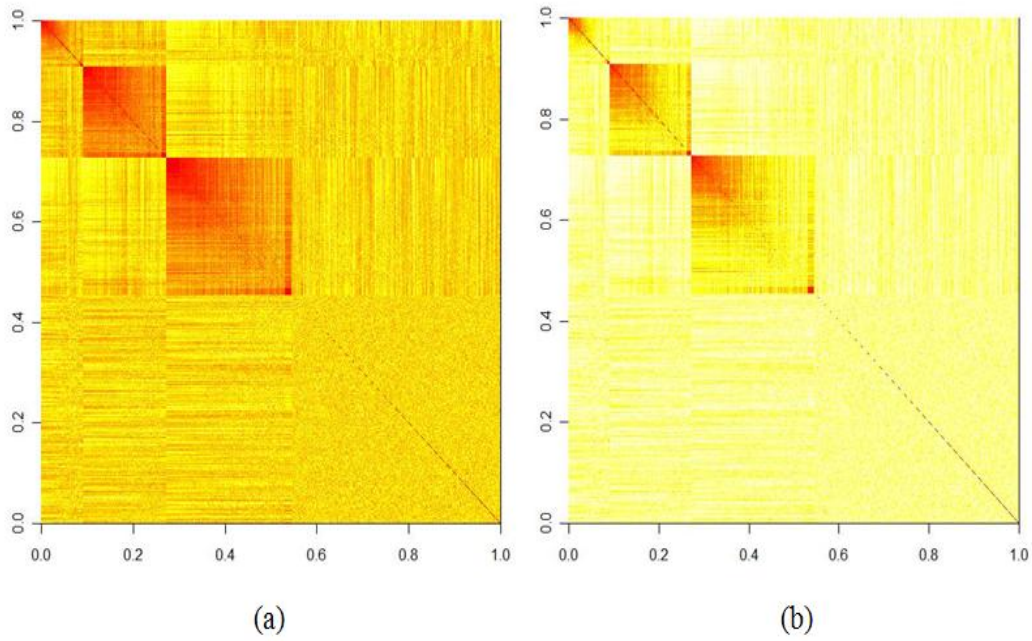

Figure 2.3: Color picture of correlaton matrix. (a) Pearson correlation coefficients (b) Corrected correlation coefficients by target estimation and stochastic approximation

2.5 Discussions

The idea of S.A.D originates from the estimation issue that appear in microarray data analysis. As mentioned in section 2.3, we need to estimate the common distribution of the standard deviation of all genes. We mentioned two algorithms in section 2.3.1.

It was seen that the old algorithm is not an step by step approximation. It is not like S.A.D. which can produce a series of distribution functions t approximate true distribution. Theoretically, we can always achieve better estimator by S.A.D than by the old algorithm though the difference between two estimators could be small.

Similarly, the old algorithm can also be applied to correlation matrix estimation. In this case, the improvement of S.A.D is bigger than the prevous case especially when our data size is very big and data shows many clusters.

Table 2.1: Difference Between Estimated Correlation and True Correlation

Interations	*mean (|estimated correlation - true correlation |)*
01	0.1217543
11	0.1041911
21	0.0918061
31	0.0829661
41	0.0763751
51	0.0712563
61	0.0671681
71	0.0639897
81	0.0613384
91	0.0591308
100	0.0574079
101	0.0572951
111	0.0557598
121	0.0544193
131	0.0532888
141	0.0523036
151	0.0514560
161	0.0507081

Generally, better perfermance means more computing time and S.A.D is not an exception. So the old algorithm is still a good alternative way to estimate distribution or correlation matrix if less time is more important in some cases.

Chapter 3

Improved Conditoinal T Approach to Identify Differentially Expressed Genes

Abstract Amaratunga and Cabrera (2003)(2007) proposed a conditional t suite of tests (Ct) for identifying differentially expressed genes in a microarray experiment. But the correlation between the mean and the variance of gene expressions is very strong in raw data. Although in many cases, the relationship is greatly reduced after taking transformation, if the correlation exists, we need deal with it. So we developed improved conditional t test to take consideration of such situations.

When the mean and the variance are independent, improved conditional t tests give us similar results as Ct. While the mean and the variance are correlated, improved Ct is evidently better than Ct in the sense that it gains more power and identifies more significantly differentially expressed genes.

3.1 Introduction

In contrast to one gene per experiment or tens of genes per experment, DNA microarrays can simutaneously measure the expression profiles of thousands of genes, often the entire repertoire of a cell population or tissue under investigation. This technology is a powerful tool to help scientists study diverse biological systems. And it is increasingly applied to address a wide range of biological questions.

A major and popular statistical analytic task in microarray data analysis is to identify significantly differentially expressed genes across two or more conditions.

Why do we need to study gene differentail expression? Because differential gene expression leads to altered cell states. Mutations of genes could cause diseases which are called generic diseases. For example,. So one experiment could be conducted to identify genes associated with the disease process so that the development of drugs can target to these genes. The simplest and most common case is a comparison between the gene expression profiles of two different experimental conditions, such as control and treatment.

Early gene expression studies declared a gene differentially expressed if its fold increase or fold decrease over a background expression level exceeded a specified cutoff, which is the simplest heuristic rule. For example, in their seminal paper on using microarrays to study gene expression in Arabidopsis thaliana, Schena et al. (1995) declared a gene differentially expressed if its expression level exhibited a fivefold difference between the two mRNA smaples. Also DeRisi et al. (1997) looked for two-fold induction of gene expression compared to baseline and in Iyer et al.(1999), genes were selected if their expression level deviated from that in control by at least a factor of 2.20 in at least two of the samples from specific cells. This approach has been criticized because it relies on fold change alone and ignores the variablity of estimates. Genes with high variablity have a larger chance to have a large fold change than genes with low variablity. Therefore, it is possible that a gene shows a five-fold change but it not significant because its expression level measurements have high variablity while a gene shows onefold change and it is significant statistically and biologically because it has low variability.

Besides heuristic rules, data mining tools such as classification appraches and clustering methods are commonly used for identifying differentially expressed genes too. for an instance, Xiong et al.(2001) identified indicator genes based on classification errors by feature wrappers (including linear discriminant analysis, logistic regression and support vector machines). Yuan and Kendziorski(2006)

proposed a unified approach for simultaneous gene clustering and differential expression identification. The clustering approach they used is lognormal-normal model(LNN) based clustering approach and AIC (Akaike information criterion), BIC (Bayesian information criterion), HQ (the criterion proposed by Hannan and Quinn (1979)) and TC (fixed cluster number at the true value) are used to select the number of clusters.

Another most widely used tools applied to micaroarray analysis are the probabilistic approaches which include non-parametric approaches and parametric approaches. Some statisticians suggest using non-parametric methods considering a microarray data often contains many noises and may be not normally distributed. Raychaudhuri et al.(2000) and Tsodikov et al. (2002) applied rank-transformed data to analyze microarray data. Dudoit et al.(2002a) used a nonparametric t-test with family wise error rate corrected p-values. Chambers et al.(1999) used the Mann-Whitney-Wilcoxon rank sum test in the analysis of microarray data from a study of human cytomegalovirus infection. Park et al(2001) scored genes based on the number of permutations of expression values required to make that gene into a perfectly discriminating marker, where all high expression values belong to one group of experiments and all low expression values belong to the other group. Significance of scores was assessed based on column permutations of the data set and comparison of the distribution of scores from permuted data to that of the original data. Other investigators used similar approaches, but looked for genes with high correlation to an idealized expression pattern that perfectly discriminates between two groups; they determined statistical significance from repeating the analysis on permuted data(Galtiske et al.(1999); Golub et al.(1999)). Troyanskaya et al.(2002) compared three model-free approaches: (1) a nonparamatric t test (2)a rank sum test and (3) a heuristic method based on high Pearson correlation to a perfectly differentiating gene and claimed that (3) was the best among the three approaches.

Although those non-parametric tests do not depend on strong distributional assumptions holding to be valid and can be used in a wide range of situations, they are less powerful than their parametric counterparts. Their p-values tend to be higher so it is harder to detect real statistically significant dfferences. When the sample size is large, the difference in power is not evident. But when sample size is small, as in typical microarray experiments, non-parametric tests have very little power to detect differences.

The parametric approaches use prabalilistic inference based on a specific data model. For example, Newton et al.(2001) identified differentially expressed genes by posterior odds of change based on a hierachical Gamma-Gamma-Bernouli model of expresion ratios. Long et al.(2001) used analysis of variance (ANOVA) based on a bayesian estimate of variance among experiment replicates with a Gaussian model for expression measurement. And a hierachical Bayesian modeling framework with Gaussian gene-independent models combined with a t-test (Baldi and Long 2001). Pan(2002) used a mixture modeling approach that estimates the distribution of t-statistic-type scores using normal mixture models and compared it with two parametric approaches, including a regular t-test.

If only two groups in comparison, two sample t test is the most basic one among all these parametric approaches. But with small samples, the t test tends to pick up significant findings at a higer rate form among the genes with low sample variance than from aong the genes with high sample variance because of the strong correlation between t statistic and the standard error estimate. This property of t test leads to a high false positive rate for genes whose variablity is low and a high false negative rate for genes whose variablity is high since the sample sizes used in microarray experiments are typically very small. There are some proposals to overcome this drawback of t test pubilished in the microarray data analysis literature. One was suggested by Tusher et al. (2001). Their significance analysis of macioarrays (SAM) method selected a constant, called fudge factor to

add to the denominator of the t test statistic and used permutations of repeated measurements to estimate the false discovery rate of differentially expressed genes. Broberg (2002) proposed an alternative method for estimating the fudge factor. For various values, the value that corresponds to the point on the ROC curve (a relationship between the false negative rate and the false negative rate) that is nearest the origin is chosen as the fudge factor.

Amaratunga and Cabrera(2003) proposed a conditional t (Ct) suite of tests to overcome the shortcomings of t tests. They estimated the distribution of t test statistic conditional on standard error under null hypothesis and produced a critical envelope instead of a critical value to decide which genes are up-regulated or down-regulated. According to simulation results, Ct is slightly better than SAM on both simulated data and real data. Now we extend Ct to handle the situation when the correlation between the mean and the variance of gene expressions exist even after data transformation or normalization. Figure 2.1 plots the sample standard deviations (s) vs sample means (x) on nomalized mice data in the R package DNAMR which shows the dependence between mean and variance.

This chapter is organized as follows. Section 3.2 reviews the Ct test. Section 3.3 introduces improved Ct test and shows its asymptotical properties. Section 3.4 shows the simulation results of Improved Ct test and its better performance compared to Ct. The proofs of theorems in section 3.3 are provided in section 3.5. Finally, section 3.6 discusses our findings and conclusions of chapter 3.

3.2 Conditional t test

In this section, we provide the procedures of conditional t(Ct) tests, which was proposed by Amaratunga and Cabrera (2003), and we compare the performance of conditional t test with the standard t test and SAM.

First we look at the model: $X_{gij} = \mu_{gj} + \sigma_g \epsilon_{gij}$, where X_{gij} is log transformed

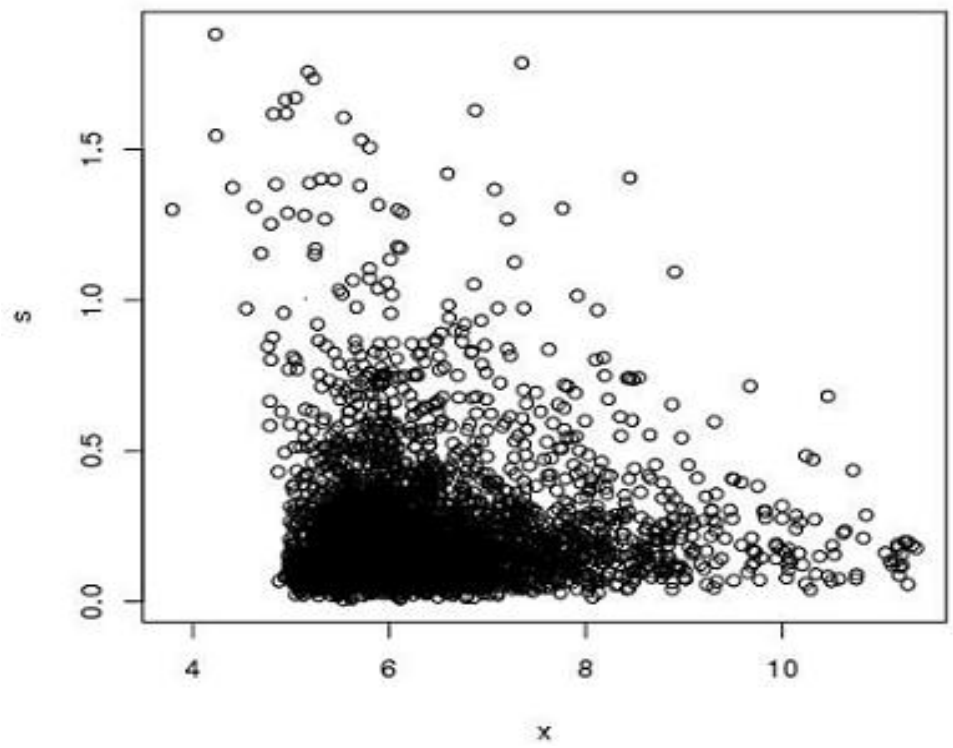

Figure 3.1: mice data: sample standard deviations (s) vs sample means (x)

and suitably normalized intensities, μ_{gj} is the mean of the gth gene in the jth group, and σ_g^2 is the variance of the gth gene. Also, g(g=1...G) indexes the genes on the microarray, j(j=1,2) indexed the groups, and i (i=1...n_j) indexes the objects.

They produced a critical envelope instead of a critical value based on the estimated distribution of T_g conditioned on s_g. The procedure is comprised of two steps (Amratunga and Cabrera (2003)(2007)):

Step1: Estimate F_σ, where F_σ is the distribution of σ_g and F_σ is the same for all the groups and all the genes. We use S.A.D presented in chapter 2 to estimate F_σ. The estimator, say $F_\sigma(x)$, will be used in the following step to generate the standard deviations of the gene populations.

Step2: Estimate the conditional distribution of $T_g|s_g$, and then estimate the values $t_\alpha(s_g)$ for a few $\alpha's$.

a) Generate a null distribution for the data by substracting the sample means

and dividing by the standard deviations.

b) Resample from the null distribution of x and multiply each sample by a σ generated from $F_\sigma(x)$. Repeated this 10,000 times and in this way, obtain 10,000 pairs of samples. From each pair of samples, calculate a value for the pooled sample standard deviation and the two sample t statistic, namely s_g and t_g for $g = 1, \ldots, 10,000$.

c) Estimate $t_\alpha(s_g)$ according to $P(|T| > t_\alpha(s_g)|s_g) = \alpha$ using a quantile regression estimate for t_g verse s_g and estimate the regression quantile curve for the $(1-\alpha)^{th}$ quantile. A rough but effective way to estimate the regression quantile curve is to split the 10,000 points into 100 groups with 100 points in each group sorted by s_g and calculate the $1-\alpha$ quantile for each group which is called $t_{(j)}$. Then the group medians for s_g, called $s_{(j)} j = 1, \ldots, 100$, are calculated. Last, estimate $t_\alpha(s_g)$ by fitting a smoother such as lowess or a smoothing spline to $t_{(j)}$ versus $s_{(j)}$. It is recommended to take the log of $t_{(j)}$ and $s_{(j)}$ first before estimating the quantile function.

Amratunga and Cabrera (2003)(2007) compare Ct to tradional t test and SAM based on simulated data and real data. Results show that the performance of Ct is slightly better than SAM, and much better than standard t test.

3.3 Improved conditional t test

When the variance (σ^2) and the mean (μ) of gene expressions are correlated, a straightforward extention of Ct is to estimate the joint distribution of σ and μ and get the distribution of T_g conditioned on joint (σ, μ). But computationally it needs using two-dimensitional smoothers and inverting two-dimentional functions, which is not easy. An alternative way is to split data by ordered mean values and apply Ct to each of the subsets. In this section, first we introduce Improved Conditional t test and then show its asympotical properties and simulation results.

3.3.1 Procedures of improved conditional t test

Interate our model: $X_{gij} = \mu_{gj} + \sigma_g \epsilon_{gij}$. g=1...G; j=1,2; and i=1...n_j.

We show three methods here:

Method 0 (Conditional t test). Apply Ct test on the whole data, and genes fall outside the critical curve $h(s_g)$ are said to be significant.

Let G be the number of genes and let $\bar{x}_{(1)}$,$\bar{x}_{(2)}$,...,$\bar{x}_{(G)}$ be the order statistics of $\bar{x}_{(g)}$, where $\bar{x}_{(g)}$ is the mean intensities of gene g. We split the data to K blocks by the values of $\bar{x}_{(g)}$ and let G_k denote the number of genes in the k^{th} block. Then the intervals of mean gene expressions of K blocks are $[\bar{x}_{(1)},\bar{x}_{(G_1)})$,$[\bar{x}_{(G_1)},\bar{x}_{(G_1+G_2-1)})$,...,$[\bar{x}_{(G-G_K+1)},\bar{x}_{(G)}]$. In practice, we set $G_k = n_b$ where n_b is a fixed value. Since $\bar{x}_{(1)}$ is the minimum and $\bar{x}_{(G)}$ is the maximum, block $[\bar{x}_{(1)},\bar{x}_{(G_1)})$ is equivalent to block $(-\infty,\bar{x}_{(G_1)})$ and similarly, block $[\bar{x}_{(G-G_K+1)},\bar{x}_{(G)}]$ is equivalent to block $[\bar{x}_{(G-G_K+1)},+\infty)$ for genes that belong to the two blocks.

Method 1. Apply Ct test on the k^{th} block to estimate a critical curve $h_k^1(s_g)$, and genes in the k^{th} block that fall outside $h_k^1(s_g)$ are said to be statistically significant for k=1,2,...,K.

Method 2. (Improved Conditional t test) It contains three steps:

1) Apply Ct test on the first 2 blocks to estimate a critical curve $h^1(s_g)$, and genes in the first block that fall outside $h^1(s_g)$ are said to be statistically significant.

2) Apply Ct test on the $(k-1)^{th}$, k^{th} and $(k+1)^{th}$ block to estimate a critical curve $h_k(s_g)$, and genes in the k^{th} block that fall outside $h_k(s_g)$ are said to be statistically significant for k=2,...,(K-1).

3) Apply Ct test on the last 2 blocks to estimate a critical curve $h_K(s_g)$, and genes in the last block that fall outside $h_K(s_g)$ are said to be statistically significant.

3.3.2 Properties

In subsection 3.3.1, we describe three methodologies for identification of significantly differetially expressed genes, here we explore the asympotical properties of these methods.

Some notations:

x: random variables representing the mean intensity;

s: random variables representing the pooled standard error estimate;

t: random variables representing the observed t-statistic;

f(t,s,x): joint probability density function of t s and x;

f(t): marginal distribution of t;

t_α: critical t value which satisfies $\int_{t_\alpha}^{\infty} f(t) = \alpha$.

First we explore the relationship between Conditional t test and standard two sample t test, that is:

Theorem 3.3.1: For Conditional t test (Method 0), as $n_1 \to \infty, n_2 \to \infty$, $h(s_g) \to t_\alpha$.

This theorem points out that Ct test goes to general two sample t test as the group sizes go to infinity.

Lemma 3.3.2: In Method 1 and Method 2,

$\bar{x}_{(G_1+...+G_{k-1}+G_k)} - \bar{x}_{(G_1+...+G_{k-1}+1)} \to 0$, as $G_k \to 0, K \to \infty, \frac{G_k}{G} \to 0$, for $k = 1, ..., K$.

Theorem 3.3.3:(levels of three tests)

(A) Ct (Method 0) is a level α test. (B) Method 1 is a level α test.

(C) Improved Ct (Method 2) is a level α test asymptotically.

Theorem 3.3.4: If μ_{gj} is asymptotically independent of σ_g^2, then three methods are equivalent.

The variance and the mean can not be completely independent, but if the correlation is very weak, we can still use Ct to save computing time. If computing

is not a problem, then Improved Ct is recommended since it is better than Ct in any case.

3.3.3 Simulation results

To compare the performance of Improved Ct and Ct, we apply these two procedures to simulated datasets.

Senario 1: First we simulate $X_{gij} \sim NID(\tau_g, 1)$, with 10000 genes and 4 samples in each group. And $\tau_g = \delta$ for g=1,...,1000 and $\tau_g = 0$ otherwise. So in this senario, 1000 genes are set up to be differentially expressed.

For various δ ($\delta = 1$ or 2) and α ($\alpha = 0.01$ or 0.05), the number of statistically significant genes (g_{sig}) at level α, the number of false discovered genes (f_{sig}) and pFDR by Ct and Improved Ct with different block size G_i are recorded in Table 3.1. These results are the average of 500 simulations. Blocka are evenly splitted (except for the last block) in Improved approach. In detail, block size G_k means that the block size of the first (K-1) group, where $K = \left\lfloor \frac{G}{G_i} \right\rfloor$, is chosen to be G_k and the block size of the K^{th} group, i.e. the last group, is $G - (K-1) * G_k$.

From Table 3.1, not surprisely, pFDR is much lower at $\delta = 1$ then at $\delta = 2$ given all the other conditions are same because as the true difference increases, methods performs better in general. For $\alpha = 0.01$, except the highest pFDR goes to Improved Ct with block size 100, the smallest block size, for both δ and smaller block sizes tend to produce higher pFDR if $\delta = 2$, pFDR of Improved Cts and Ct are comparable at either δ. While when $\alpha = 0.05$, things are a little different; in this case, no matter $\delta = 1$ or $\delta = 2$, there is no much difference among g_{sig} or f_{sig} or pFDR of Ct and Improved Ct with various block size.

Table 3.2 and Table 3.3 shows the 20 most significant genes calculated by Ct and Improved Ct in one simulation for $\delta = 1$ and $\delta = 2$ respectively. Numbers in two tables are the labels of genes. The genes whose labels are larger than 1000 are falsely declared genes since only the first 1000 genes are truely differentially

Table 3.1: (Senario I) g_{sig}, f_{sig} and pFDR calculated by Ct and Improved Ct $X_{gij} \sim NID(\tau_g, 1)$, $\tau_g = \delta \cdot 1_{(1 \le g \le 1000)}$, $G = 10,000$. $n_1 = n_2 = 4$

		$\alpha = 0.01$			$\alpha = 0.05$		
$\delta = 1$		g_{sig}	f_{sig}	pFDR	g_{sig}	f_{sig}	pFDR
	Ct	208	86	41.3%	823	503	61.1%
	100	278	135	48.6%	877	564	64.3%
	150	238	105	44.1%	865	549	63.4%
	200	206	78	37.9%	794	482	60.7%
	250	214	96	44.9%	818	500	61.1%
	300	222	96	43.2%	839	527	62.8%
G_i	350	232	104	44.8%	820	514	62.7%
	400	245	109	44.5%	880	556	63.2%
	500	239	115	48.1%	875	556	63.5%
	600	219	96	43.9%	825	515	62.4%
	700	204	83	40.7%	836	516	61.7%
	800	229	99	43.2%	855	533	62.3%
	900	231	99	42.9%	846	523	61.8%
$\delta = 2$							
	Ct	679	85	12.5%	1324	500	37.8%
	100	684	116	17.0%	1287	516	40.1%
	150	724	119	16.4%	1378	553	40.1%
	200	713	113	15.8%	1358	531	39.1%
	250	702	118	16.8%	1347	528	39.2%
	300	706	107	15.2%	1359	531	39.1%
G_i	350	719	115	16.0%	1386	557	40.2%
	400	711	107	15.0%	1359	524	38.6%
	500	690	88	12.8%	1342	511	38.0%
	600	709	96	13.5%	1346	517	38.4%
	700	692	92	13.9%	1356	527	38.9%
	800	683	83	12.2%	1320	498	37.7%
	900	692	87	12.6%	1332	512	38.4%

Table 3.2: Simulation I: 20 most significant genes in one simulation. $G = 10,000$, $n_1 = n_2 = 4$, $X_{gij} \sim NID(\tau_g, 1)$, $\tau_g = 1_{(1 \leq g \leq 1000)}$

		G_i								
	Ct	100	200	300	400	500	600	700	800	900
1	943	5881	4272	943	943	943	943	943	943	943
2	686	9195	943	686	551	686	939	2056	686	370
3	518	4272	9059	2056	785	518	686	686	518	686
4	2056	6222	2056	518	796	2431	518	518	2056	785
5	785	686	686	9059	518	796	370	284	551	518
6	649	4132	683	785	2056	655	683	785	9059	9059
7	9059	943	68	370	370	284	68	2431	649	2056
8	370	518	649	504	686	785	504	649	370	68
9	551	8160	785	68	9059	9059	649	370	785	551
10	68	655	518	551	504	807	785	436	796	649
11	605	504	771	683	683	7595	796	605	284	317
12	796	730	504	317	839	771	655	68	605	284
13	807	807	796	655	436	649	605	9059	655	605
14	655	939	448	796	68	730	807	796	683	796
15	317	785	317	436	655	504	19	655	504	655
16	436	9858	839	807	284	683	9059	317	807	436
17	284	9059	730	649	7595	839	284	839	317	504
18	683	4610	8160	284	317	551	2056	551	771	807
19	771	7595	19	860	19	605	884	7595	436	683
20	504	68	835	839	835	2116	730	730	839	835
#non	2	10	4	2	3	3	2	4	2	2

Table 3.3: Simulation II: 20 most significant genes in one simulation. $G = 10,000$, $n_1 = n_2 = 4$, $X_{gij} \sim NID(\tau_g, 1)$, $\tau_g = 2 \cdot 1_{(1 \leq g \leq 1000)}$

		G_i										
	Ct	100	150	200	250	300	400	500	600	700	800	900
1	943	4932	796	943	370	943	4232	943	943	943	943	943
2	686	8160	943	518	943	686	588	683	686	68	686	370
3	518	966	370	686	452	518	370	686	518	683	785	68
4	370	452	730	835	518	655	68	518	649	370	370	686
5	785	68	68	683	68	730	943	19	605	785	518	518
6	68	686	884	588	686	796	551	504	436	19	284	436
7	649	317	518	317	551	860	436	302	796	504	771	785
8	284	889	686	436	655	839	966	317	317	401	209	649
9	551	518	504	209	796	828	284	796	209	436	452	807
10	683	236	785	655	966	793	807	807	655	649	401	551
11	436	943	860	370	785	649	924	401	504	209	68	605
12	807	655	649	796	29	360	317	142	683	142	655	796
13	796	796	61	19	554	361	309	612	370	605	796	655
14	605	501	655	771	771	370	686	889	785	884	839	884
15	504	504	175	302	730	924	785	370	551	518	807	284
16	655	436	137	68	605	785	518	655	401	686	436	839
17	771	257	360	649	860	58	839	884	68	924	317	683
18	317	835	605	828	257	309	884	551	771	796	649	966
19	839	839	924	142	61	504	448	785	730	284	551	730
20	730	248	614	939	835	424	771	209	284	309	730	504
#non	0	2	0	0	0	0	1	0	0	0	0	0

expressed. The last row displays the falsely discovered genes among top 20 genes. From Table 3.2, it is easily seen that Ct and Improved Ct with different block sizes share common declared significantly differentially expressed genes including truely differentially expressed genes and falsely discovered genes. There are 10 falsely discovered genes among top 20 when block size is 100 and 2 to 6 falsely discovered genes in other cases. It partially implies worse performance of Improved Ct with blcok size 100, which is consistent with the results shown in Table 3.1. Similar phenomenon exist in Table 3.3. But since the true difference has increased from 1 to 2, there is few falsely discovered genes among top 20 significant genes.

Senario 2: we simulate $X_{gij} \sim NID(\tau_{gj}, s_g^2)$, with G genes and 4 samples in each group. And $\tau_{g1} \sim U(0, 100)$, $\tau_{g2} = \tau_{g1} + \delta$ for g=1,...,1000 and $\tau_{g2} = \tau_{g1}$ otherwise. $s_g^2 = \tau_{g1} + \chi_3^2$. In this senario, still 1000 genes are set up to be differentially expressed.

Under Senario 2, for various δ and G, the number of statistically significant genes (g_{sig}) at level α, the number of false discovered genes (f_{sig}) and pFDR by Ct and Improved Ct with different block size G_k are recorded in Table 3.4 - Table 3.6. These results are the average of 500 simulations. Blocks splitting method in Improved Ct are the same as previous simulation under Senario 1.

From Table 3.4, when $\alpha = 0.01$, the pFDRs of all Improved Ct are uniformly less than that of Ct while the number of statistically significant genes ($g_{sig}s$) and the number of false discovered genes ($f_{sig}s$) are greater than those of Ct. And as the block size of Improved Ct increases, both g_{sig} and f_{sig} tend to decrease. When $\alpha = 0.05$, the pFDRs of Improved Ct and Ct are very close. Results are similar in Table 3.5 and Table 3.6.

Table 3.4: g_{sig}, f_{sig} and pFDR calculated by Ct and Improved Ct. $G = 10000$, $n_1 = n_2 = 4$, $X_{gij} \sim NID(\tau_{gj}, s_g^2)$, $\tau_{g1} \sim U(0, 100)$, $\tau_{g2} = \tau_{g1} + \delta \cdot 1_{(1 \le g \le 1000)}$, $s_g^2 = \tau_{g1} + \chi_3^2$

		$\alpha = 0.01$			$\alpha = 0.05$		
$\delta = 5$		g_{sig}	f_{sig}	pFDR	g_{sig}	f_{sig}	pFDR
	Ct	129	78	60.5%	606	430	71.0%
	100	250	139	55.6%	798	555	69.6%
	150	226	109	58.2%	793	551	69.5%
	200	230	118	51.3%	785	546	69.6%
	250	229	117	51.1%	793	542	68.3%
	300	213	101	47.4%	794	548	69.0%
	350	215	104	48.3%	787	546	69.3%
	400	218	108	49.5%	769	529	68.8%
G_i	500	201	100	49.8%	764	519	67.9%
	600	203	101	49.8%	755	520	68.9%
	700	202	98	48.5%	751	512	68.2%
	800	180	87	48.3%	716	489	68.3%
	900	189	89	47.0%	726	499	68.7%
	1000	175	84	48.0%	725	494	68.1%
	1100	173	86	49.7%	719	496	69.0%
	1200	171	86	50.3%	706	483	68.4%
$\delta = 10$							
	Ct	333	89	26.7%	953	449	47.1%
	100	513	136	26.5%	1147	583	50.8%
	150	479	119	24.8%	1124	570	50.7%
	200	462	118	25.5%	1086	539	49.6%
	250	467	113	24.2%	1136	573	50.4%
	300	454	102	22.5%	1084	535	49.4%
	350	475	110	23.1%	1082	541	50.0%
	400	467	114	24.4%	1095	536	48.9%
G_i	500	457	106	23.2%	1068	516	48.3%
	600	438	93	21.2%	1081	529	58.9%
	700	447	100	22.3%	1068	521	48.8%
	800	441	97	22.0%	1075	517	48.1%
	900	418	85	20.3%	1040	491	47.2%
	1000	421	91	21.6%	1045	497	47.6%
	1100	406	83	20.4%	1030	484	47.0%
	1200	403	75	18.6%	1032	486	47.1%

Table 3.5: g_{sig}, f_{sig} and pFDR calculated by Ct and Improved Ct $G = 20000$, $n_1 = n_2 = 4$, $X_{gij} \sim NID(\tau_{gj}, s_g^2)$, $\tau_{g1} \sim U(0, 100)$, $\tau_{g2} = \tau_{g1} + \delta \cdot 1_{(1 \le g \le 1000)}$, $s_g^2 = \tau_{g1} + \chi_3^2$

		$\alpha = 0.01$			$\alpha = 0.05$		
$\delta = 5$		g_{sig}	f_{sig}	pFDR	g_{sig}	f_{sig}	pFDR
	Ct	216	153	70.8%	1118	882	78.9%
	100	348	225	64.7%	1389	1132	81.5%
	150	354	231	65.3%	1427	1165	81.6%
	200	344	222	64.5%	1382	1118	80.9%
	250	326	206	63.2%	1394	1132	81.2%
	300	309	193	62.5%	1355	1092	80.1%
	350	307	192	62.5%	1326	1069	80.6%
	400	305	189	62.0%	1326	1070	80.7%
G_i	500	306	195	63.7%	1343	1078	80.2%
	600	293	183	62.5%	1325	1066	80.5%
	700	284	181	63.7%	1282	1022	79.7%
	800	284	185	65.1%	1303	1045	80.1%
	900	269	174	64.7%	1281	1028	80.2%
	1000	273	170	62.3%	1270	1013	79.8%
	1100	262	167	63.7%	1251	1001	80.0%
	1200	264	172	65.2%	1274	1019	80.0%
	1300	253	159	62.8%	1271	1014	79.8%
	1400	237	155	65.4%	1246	998	80.1%
	1500	247	164	66.4%	1274	1024	80.4%
	1600	252	169	67.1%	1253	1001	79.9%
	1700	233	158	67.8%	1221	973	79.7%
	1800	239	164	68.6%	1232	977	79.3%
	1900	234	162	69.2%	1212	962	79.4%
	2000	240	164	68.3%	1238	991	80.0%

Table 3.6: g_{sig}, f_{sig} and pFDR calculated by Ct and Improved Ct $G = 20000$, $n_1 = n_2 = 4$, $X_{gij} \sim NID(\tau_{gj}, s_g^2)$, $\tau_{g1} \sim U(0, 100)$, $\tau_{g2} = \tau_{g1} + \delta \cdot 1_{(1 \le g \le 1000)}$, $s_g^2 = \tau_{g1} + \chi_3^2$

		$\alpha = 0.01$			$\alpha = 0.05$		
	$\delta = 10$	g_{sig}	f_{sig}	pFDR	g_{sig}	f_{sig}	pFDR
	Ct	402	148	36.8%	1438	855	59.5%
	100	613	242	39.5%	1707	1137	66.6%
	150	624	226	36.2%	1724	1132	65.7%
	200	602	214	35.5%	1714	1123	65.5%
	250	562	186	33.1%	1682	1085	64.5%
	300	564	189	33.5%	1636	1058	64.7%
	350	583	204	35.0%	1654	1070	64.7%
	400	554	180	32.5%	1645	1057	64.2%
G_i	500	562	188	33.4%	1649	1061	64.3%
	600	559	176	31.5%	1631	1045	64.1%
	700	566	193	34.1%	1649	1060	64.3%
	800	565	187	33.1%	1645	1055	64.1%
	900	554	180	32.4%	1627	1039	63.9%
	1000	539	172	31.9%	1571	992	63.1%
	1100	552	182	33.0%	1570	987	62.9%
	1200	552	182	33.0%	1618	1032	63.8%
	1300	551	183	33.2%	1623	1040	64.1%
	1400	519	156	30.1%	1576	994	63.0%
	1500	532	160	30.1%	1588	1006	63.4%
	1600	532	170	32.0%	1586	1008	63.6%
	1700	533	170	31.9%	1572	991	63.0%
	1800	517	159	30.8%	1568	985	62.8%
	1900	513	154	30.0%	1532	948	61.9%
	2000	506	152	30.0%	1541	967	62.8%

Comparison: To compare Ct, Improved Ct with traditional t test, FDR50, the proportion of the top 50 genes that are not within the differentially expressed genes, is used as a criterion in the following two situations: first, we simulate $X_{gij} \sim NID(\tau_g, 1)$, with 5000 genes and 4 samples in each group, where $\tau_g = \delta$ for g=1,...,100 and $\tau_g = 0$ otherwise. FDR50 curves from 500 simulations performed for each of 5 different values of δ between 0.5 and 2.5 are shown in Figure 3.2. The block size we use in Improved Ct is 500. Then we simulate $X_{gij} \sim NID(\tau_{gj}, s_g^2)$, with 5000 genes and 4 samples in each group. And $\tau_{g1} \sim U(0, 100)$, $\tau_{g2} = \tau_{g1} + \delta$ for g=1,...,100 and $\tau_{g2} = \tau_{g1}$ otherwise. $s_g^2 = \tau_{g1} + \chi_3^2$. Figure 3.3 shows FDR50 curves from 500 simulations performed for each of 7 different values of δ between 4 and 16.

It is easily seen that when the mean and the variance are independent, Ct is comparable to Improved Ct and both are much better than traditional t test. While when there is strong dependence between the mean and the variance, Improved Ct is much better than the other two.

We also used Khans pediatric tumor dataset (Khan et al (2001) which contains 2308 genes to check the performance of Ct, Improved Ct and traditional t. From this dataset, we chose columns 6 to 13 corresponding to 8 patients with Ewing tumors since these columns appear to have no differentially expressed genes and we can add a δ to randomly selected genes in the treatment group so that these genes are artificially differentially expressed. We split the subset into a control group (columns 6 to 9) and a treatment group (columns 10 to 13). Each time, we added δ to 100 randomly selected genes for the treatment group. The average FDR50 of 500 times calculated by three methods corresponding to 7 values of δ between 0.2 and 0.8 are shown in Figure 3.3. The block size we choose for Improved Ct is 350 which results in six blocks.

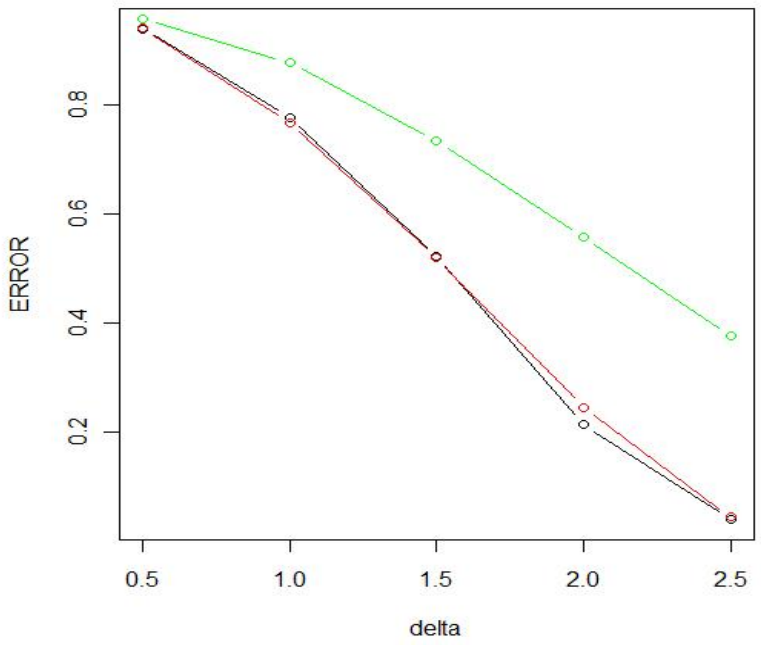

Figure 3.2: FDR50 Curves: Red: Improved Ct (block size 500); Black: Ct; Green: t test approach

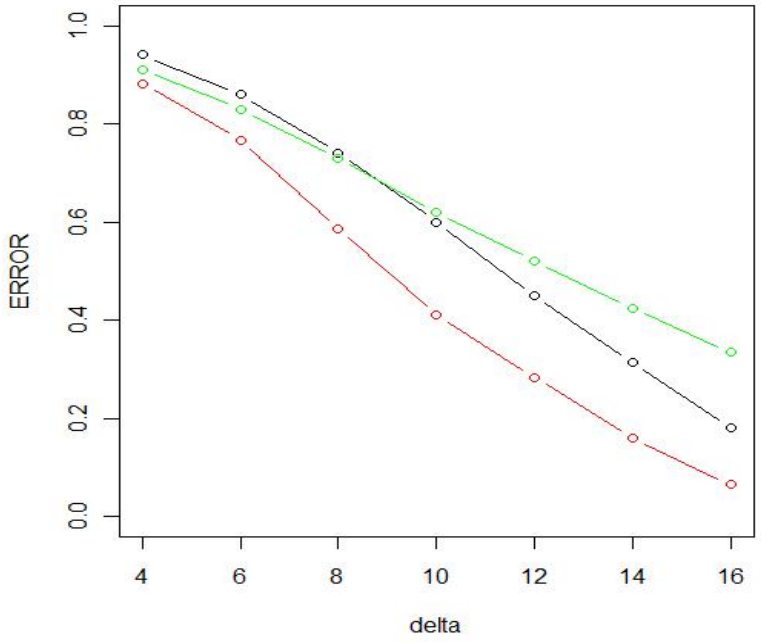

Figure 3.3: FDR50 Curves: Red: Improved Ct (block size 500); Black: Ct; Green: t test approach

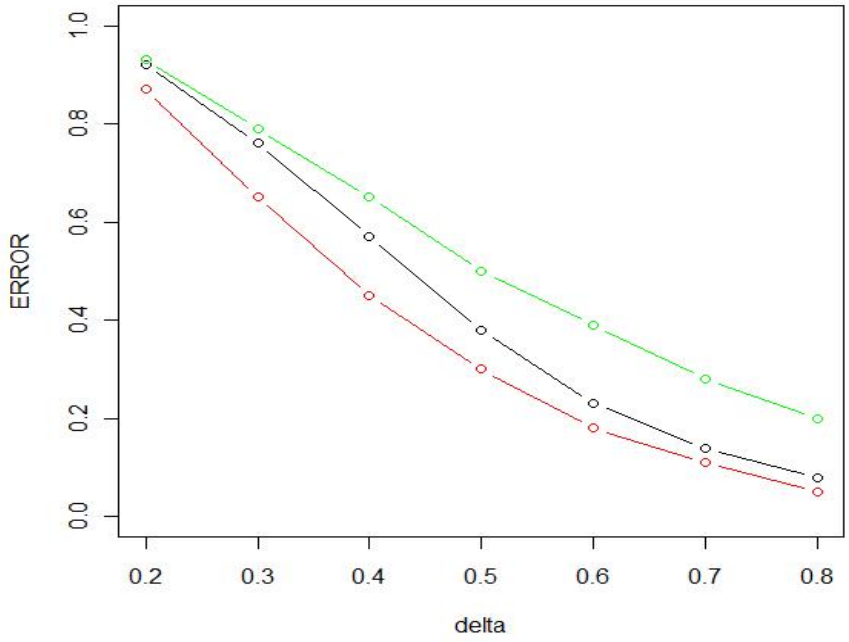

Figure 3.4: FDR50 Curves: Red: Improved Ct (block size 350); Black: Ct; Green: t test approach

3.4 Proofs

In this section, we provide the detailed proofs of theorems in section 3.3.

Theorem 3.3.1: For Conditional t test (Method 0), as $n_1 \to \infty, n_2 \to \infty$, $h(s_g) \to t_\alpha$.

Proof: As $n_1 \to \infty$ and $n_2 \to \infty$, by the Law of Large Number,

it is easily seen that $s_g \to \sigma_g$ a.s. which implies $\frac{\sigma_g^2}{s_g^2} \to 1$ a.s.

By the central limit theorem, we have $T_g|s_g = \frac{X_{g2}-X_{g1}}{s_g\sqrt{\frac{1}{n_1}+\frac{1}{n_2}}} \to N(0,1)$

Since $t_\alpha(s_g)$ is defined by $P(|T| > t_\alpha(s_g)\,|s_g; H_0) = \alpha$,,

$T_\alpha(s_g) \to \varphi_\alpha$, $t_\alpha \to \varphi_\alpha$

where φ_α is the α quantile of the standard normal distribution.

Hence, $T_\alpha(s_g) \to t_\alpha$.

Lemma 3.3.2: In Method 1 and Method 2,

$\bar{x}_{(G_1+...+G_{k-1}+G_k)} - \bar{x}_{(G_1+...+G_{k-1}+1)} \to 0$, as $G_k \to 0, K \to \infty, \frac{G_k}{G} \to 0$, for $k = 1, ..., K$.

Proof: Let $f_x(x)$ and $F_x(x)$ be the density function and distribution function of x, the mean intensity of a random selcted gene.

As $G_k \to 0, K \to \infty, \frac{G_k}{G} \to 0$, by laws of large numbers,

$\bar{x}_{(G_1+...+G_{k-1}+G_k)} \to \frac{G_1+...+G_{k-1}+G_k}{G} th$ quantile

$\bar{x}_{(G_1+...+G_{k-1}+1)} \to \frac{G_1+...+G_{k-1}+1}{G} th$ quantile

$F_x(\bar{x}_{(G_1+...+G_{k-1}+G_k)}) - F_x(\bar{x}_{(G_1+...+G_{k-1}+1)}) \to \frac{G_k-1}{G} \to 0$

$\Rightarrow \int_{\bar{x}_{(G_1+...+G_{k-1}+1)}}^{\bar{x}_{(G_1+...+G_{k-1}+G_k)}} f_x(x)\,dx \to 0$

Since $f_x(x) > 0$ a.e, we have $\bar{x}_{(G_1+...+G_{k-1}+G_k)} - \bar{x}_{(G_1+...+G_{k-1}+1)} \to 0$.

Theorem 3.3.3: (A) Ct (Method 0) is a level α test. (B) Method 1 is a level α test. (C) Improved Ct (Method 2) is a level α test asymptotically.

Proof: Part (A): Method 0 is a level α test, which is proved in Amaratunga and Cabrera (2003).

In Ct procedure, the null hypothesis is rejected if $t > h(s)$ and conditionng on s the probability of type one error is α. The overall unconditional probability of type one error is also α because:

$$\int_0^\infty \int_{t_\alpha(s)}^\infty f(t,s)dtds = \int_0^\infty \Big(\int_{-\infty}^\infty f(t,s)dt \Big) \frac{\int_{t_\alpha(s)}^\infty f(t,s)dt}{\int_{-\infty}^\infty f(t,s)dt} ds$$

$$= \int_0^\infty \int_{-\infty}^\infty f(t,s)dt\alpha ds = \alpha \int_0^\infty \int_{-\infty}^\infty f(t,s)dtds = \alpha$$

Part (B): Method 1 is a level α test.

Based on the procedure of Method 1, we have the conditional probability of type one error is α, i.e.

$$P(T > h_k^1(s_g) \,\Big|\, \bar{x}_g \in [\bar{x}_{(G_1+\ldots+G_{k-1}+1)}, \bar{x}_{(G_1+\ldots+G_{k-1}+G_k)}], s_g) = \alpha,$$

for $k = 1, 2, \cdots, K$.

Let $h(s)$ be the family of critical curves, the overall unconditional probability of Type I error is α because of the following calculation:

$$\begin{aligned}
\int_{-\infty}^\infty \int_0^\infty \int_{h(s)}^\infty f(t,s,x)\,dtdsdx &= \sum_{k=1}^K \int_{\bar{x}_{(G_1+\ldots+G_{i-1}+1)}}^{\bar{x}_{(G_1+\ldots+G_{k-1}+G_k)}} \int_0^\infty \int_{h(s)}^\infty f(t,s,x)\,dtdsdx \\
&= \sum_{k=1}^K \int_{\bar{x}_{(G_1+\ldots+G_{k-1}+1)}}^{\bar{x}_{(G_1+\ldots+G_{k-1}+G_k)}} \int_0^\infty \int_{h_k^1(s)}^\infty f(t,s,x)\,dtdsdx \\
&= \sum_{k=1}^K \int_{\bar{x}_{(G_1+\ldots+G_{k-1}+1)}}^{\bar{x}_{(G_1+\ldots+G_{k-1}+G_k)}} \int_0^\infty \alpha \int_{-\infty}^\infty f(t,s,x)\,dtdsdx \\
&= \alpha\Big(\int_{-\infty}^{\bar{x}_{G_1+G_2}} + \sum_{k=2}^{K-1} \int_{\bar{x}_{(G_1+\ldots+G_{k-1}+1)}}^{\bar{x}_{(G_1+\ldots+G_{k-1}+G_k)}} + \int_{\bar{x}_{G-G_K+1}}^{\infty} \Big) \\
&\quad \int_0^\infty \int_{-\infty}^\infty f(t,s,x)\,dtdsdx \\
&= \alpha \int_{-\infty}^\infty \int_0^\infty \int_{-\infty}^\infty f(t,s,x)\,dtdsdx = \alpha
\end{aligned}$$

(3.2)

Part (C):Improved Ct (Method 2) is a level α test asymptotically.

We will show type I error of Improved Ct $\to \alpha$, as $G_k \to 0, K \to \infty$, $\frac{G_k}{G} \to 0$, $k = 1, ..., K$.

Let $f_x(x)$ be the density function of x, the mean intensity of a random selcted gene, and $f(x\,|T > h_k(s_g))$ be the conditional density of x, given $T > h_k(s_g)$.

Let $\tilde{x} = \bar{x}_{(G_1+...+G_{k-1}+1)}$, then we have $\bar{x}_{(G_1+...+G_{k-2}+1)} = \tilde{x} - \Delta x_1$,

$\bar{x}_{(G_1+...+G_{k-1}+G_k)} = \tilde{x} + \Delta x_2$, $\bar{x}_{(G_1+...+G_{k-1}+G_{k+1})} = \tilde{x} + \Delta x_2 + \Delta x_3$

by Lemma2.3.2, we get $\Delta x_1 \to 0, \Delta x_2 \to 0, \Delta x_3 \to 0$ as $G_k \to 0, K \to \infty$, $\frac{G_k}{G} \to 0$, for $k = 1, ..., K$.

$P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(G_1+...+G_{k-1}+1)}, \bar{x}_{(G_1+...+G_{k-1}+G_k)}], s_g)$

$= P((T > h_k(s_g)|s_g)\,|\bar{x}_g \in [\tilde{x}, \tilde{x} + \Delta x_2))$

$= (P(\bar{x}_g \in [\tilde{x}, \tilde{x} + \Delta x_2)\,|T > h_k(s_g)) \cdot P(T > h_k(s_g))/P(\bar{x}_g \in [\tilde{x}, \tilde{x} + \Delta x_2))$

$= f(\tilde{x}\,|T > h_k(s_g)) \cdot \Delta x_2 \cdot P(T > h_k(s_g))/(f(\tilde{x}) \cdot \Delta x_2)$

$= f(\tilde{x}\,|T > h_k(s_g)) \cdot P(T > h_k(s_g))/f(\tilde{x})$.

Similarly, $P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(G_1+...+G_{k-2}+1)}, \bar{x}_{(G_1+...+G_k+G_{k+1})}], s_g)$

$= P((T > h_k(s_g)|s_g)\,|\bar{x}_g \in [\tilde{x} - \Delta x_1, \tilde{x} + \Delta x_2) + \Delta x_3)$

$= f(\tilde{x}\,|T > h_k(s_g)) \cdot P(T > h_k(s_g))/f(\tilde{x})$.

So $P(T\, h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(G_1+...+G_{k-1}+1)}, \bar{x}_{(G_1+...+G_{k-1}+G_k)}], s_g)$

$= P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(G_1+...+G_{k-2}+1)}, \bar{x}_{(G_1+...+G_k+G_{k+1})}], s_g)$,

for $k = 2, \cdots, (K-1)$.

With the same strategy, we can show that asymptotically,

$P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(1)}, \bar{x}_{(G_1)}], s_g) = P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(1)}, \bar{x}_{(G_1+G_2)}], s_g)$

and $P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(G_1+...+G_{K-1}+1)}, \bar{x}_{(G)}], s_g)$

$= P(T > h_k(s_g) \,\big|\bar{x}_g \in [\bar{x}_{(G_1+...+G_{K-2}+1)}, \bar{x}_(G)], s_g)$.

It is easily seen that Improved ct is a level α test asymptotically using the same way as shown in Lemma 1. What we need do is replacing $h_k^1(s)$ by $h_k(s)$.

Theorem 2.3.4. If μ_{gj} is asymptotically independent of σ_g^2, then three methods are equivalent.

Proof:

Method 0: $P(T > h(s_g)|s_g) = \alpha$.

Method 1:

$\alpha = P(T > h_k^1(s_g) \left| \bar{x}_g \in [\bar{x}_{(G_1+\ldots+G_{k-1}+1)}, \bar{x}_{(G_1+\ldots+G_{k-1}+G_k)}], s_g \right.)$

$\rightarrow P(T > h_k^1(s_g)|s_g)$.

Method 2:

$$\begin{cases} \alpha = P(T > h_1(s_g) \left| \bar{x}_g \in [\bar{x}_{(1)}, \bar{x}_{(G_1+G_2)}], s_g \right.) \rightarrow P(T > h_1(s_g)|s_g) \\ \alpha = P(T > h_k(s_g) \left| \bar{x}_g \in [\bar{x}_{(G_1+\ldots+G_{k-2}+1)}, \bar{x}_{(G_1+\ldots+G_k+G_{k+1})}], s_g \right.) \\ \rightarrow P(T > h_k(s_g)|s_g) \\ \alpha = P(T > h_k(s_g) \left| \bar{x}_g \in [\bar{x}_{(G_1+\ldots+G_{k-2}+1)}, \bar{x}_(G)], s_g \right.) \rightarrow P(T > h_k(s_g)|s_g) \end{cases}$$

so for any given s_g,

$P(T > h(s_g)|s_g) = P(T > h_k^1(s_g)|s_g) = P(T > h_k(s_g)|s_g) = \alpha$.

Let $f_{t,s}(t,s)$ be the joint distribution of t and s, then

$$\frac{\int_{h(s_g)}^{\infty} f(t,s)\,dt}{\int_{-\infty}^{\infty} f(t,s)\,dt} = \frac{\int_{h_k^1(s_g)}^{\infty} f(t,s)\,dt}{\int_{-\infty}^{\infty} f(t,s)\,dt} = \frac{\int_{h_k(s_g)}^{\infty} f(t,s)\,dt}{\int_{-\infty}^{\infty} f(t,s)\,dt}$$

It is easily seen that $h(s_g) = h_k^1(s_g) = h_i(s_g), a.s$, so the three methods are equivalent.

3.5 Discussions

We have proposed Improved Ct methodology to identify differentially expressed genes. It is an extenstion of Ct proposed by Amaratunga and Cabrera (2003)(2007). When the dependence between the mean and the variance is weak, simulation results show that pFDR of Ct and Improved Ct are very close. In this case, Ct may be considered superior to Improved Ct in the sense that Ct is computationally faster.

When the dependence between the mean and the variance is strong, Improved Ct with suitable block sizes is better than Ct. One problem is how to select block size. If the block size is too small, e.g. 100 in simulation, the standard error estimate is very biased so that the performance of Improved Ct could be worse than that of Ct and small block size means large blocks and means much computing time. As the block size becomes larger, Improved Ct will be better than Ct. But if the block size is too large, which means the number of blocks is few, the results of Improved Ct will be smilar to those of Ct note that Ct is a special case of Improved Ct where there is only one or two blocks. So the block size can not be too small or too large, one should choose moderate block size. We do not have a formulato calculate the best block size because it is data dependent which can be seen from Table 2.4: two kinds of datasets contan the same number of genes but they do not have the same best block sizes.

In our simulation, we split the data evenly so that all blocks are approximately of same number of genes. There are some other ways to split data, for example, we could let genes whose means fall in $[\bar{x}_{(1)} + (k-1)\frac{\bar{x}_{(G)} - \bar{x}_{(1)}}{K},\ \bar{x}_{(1)} + k\frac{\bar{x}_{(G)} - \bar{x}_{(1)}}{K})$ belong to the k^{th} block so that all K blocks have equall interval $\frac{\bar{x}_{(G)} - \bar{x}_{(1)}}{K}$. One can choose appropriate ways based on distribution properties of data.

Chapter 4

Improve Statistical Power for Analysis of Microarray Data Using Clustering and Variance Correction

Abstract Assessing differentially expressed genes is an important goal of microarray experiment. A common problem related to it is how to improve the power for detection of differentially expressed genes. One possible solution is to induce clustering and correlation correction to gene expression identification. On one hand, after a cluster analysis, genes performing similar functions or participanting in the same genetic pathway would congregate in the same cluster. we could choose one or more clusters we are interested in to do analysis. On the other hand, various researchers have suggested that accounting for correlation among genes could improve the power. There exists, however, three challenges when considering the the cluster pattern and correlation structure among genes: the first one is which clustering method we choose, the second one is how to reliably estimate the covariance matrix of genes and the last one is how to model the data, perform the appropriate statistical test and calculate the power. In this article, we present our methodology to tackle these problems.

4.1 Introduction

As a tremendous improvement over tedious "'one gene per experiment" paradigm, DNA microarray is the most widely used technology in biomedical research to investigate the expression patterns of thousands of genes simultaneously. This

powerful tool allows scientists to study how genes function, not only each on its own, but jointly as well.

In a typical microarray data, the number of genes is large, say a few thousands or more, and the number of samples is very small, say between 5 and 50. This characteristic of microarray data imposes challenges for statisticians.

An important goal of microarray experiment is assessing differentially expressed genes under different conditions, especially two conditions because differential gene expression leads to altered cell states. Various researchers have suggested that accounting for correlation among genes could improve the power for detection of differentially expressed genes. Intuitively, more power would be gained among highly correlated genes than independent genes given all other conditions are the same.

Cluster analysis is another major and popular statistical task in micrarray data analysis. It sorts the entirety of genes into a series of clusters so that the genes that behaved the most similarly in the experiment will be members of the same cluster, while genes that behaved differently will be members of different clusters. The rationale lies in it is that it is reasonable to expect that genes performing similar functions or operating in the same genetic pathway would behave similarly across conditions. Since the seminal paper by Eisen et al (1998), various clustering approaches have been developed in the context of microarray data such as hierarchical clustering, partitioning methods, and model-based clustering etc.

Cluster analysis is usually not used to improve power of gene differential expression idnetification although it is essential. But if we notice that microarray data contains large proportions of noises which reduces power of analysis, it is quite straightforward to do cluster analysis before more work so that we could focus on the interested clusters only.

4.2 Methodology

We only consider the simplest and most common case — a comparison between the gene expression profiles of two groups: control group and treatment group. And we assume data is suitably transformed and normalized. Let X_{gij} denote the intensity measurement for the g^{th} gene in the i^{th} microarray in the j^{th} group, where i=1,...,n_j ($n = n_1 + n_2$); j=1,2; and g=1,...,G.

The data structure is like:

$$
\begin{array}{c} \\ 1 \\ 2 \\ .. \\ G \end{array}
\begin{array}{c}
\begin{array}{cccccc} i=1, & 2,... & n_1 & i=1, & 2,... & n_2 \end{array} \\
\begin{pmatrix}
X_{111}, & X_{121},... & X_{1n_11} & X_{112}, & X_{122},... & X_{1n_12} \\
X_{211}, & X_{221},... & X_{2n_11} & X_{212}, & X_{222},... & X_{2n_12} \\
 & ... & & & ... & \\
X_{G11}, & X_{G21},... & X_{Gn_11} & X_{G12}, & X_{G22},... & X_{Gn_12}
\end{pmatrix}
\end{array}
\quad g =
$$

4.2.1 Model-Based Clustering and Correlation Matrix Estimation

Model-based clustering has been applied to microarray data by Yeung et al.(2001), McLachlan et al.(2002) and Pan et al.(2002). Some similar approaches are formulated by Holmes and Bruno (2000) and Barash and Friedman(2002). It is a partitioning method which assumes that each cluster is generated by a probability distribution. Namely, if gene g comes from the kth cluster, and let $f_k(.,.)$ be the distribution of the kth cluster, then

$x_g = (x_{g1}, x_{g2}) \sim f_k(x_{g1}, x_{g2})$,

where $x_{g1} = (x_{g11}, x_{g21}, ..., x_{gn_11})$ and $x_{g2} = (x_{g12}, x_{g22}, ..., x_{gn_22})$.

Given the prior probability, p_k (where $\sum_{k=1}^{r} p_k = 1$), of the gth gene belong to the kth cluster, an observation of gene g should follow the mixture distribution:

$x_g = (x_{g1}, x_{g2}) \sim \sum_{k=1}^{r} p_k f_k(x_{g1}, x_{g2})$,

When The distribution of $f_k(x_{g1}, x_{g2})$ is multivariate normal distribution with

parameters μ_k (mean vector) and Σ_k (covariance matrix):

$f_k(x_{g1}, x_{g2}|\mu_k, \Sigma_k) = \frac{exp\{-\frac{1}{2}(x_g-\mu_k)^T \sum_k^{-1}(x_g-\mu_k)\}}{\sqrt{det(2\pi\Sigma_k)}}$

Yeung et al.(2001) detailedly addressed various formats of covariance matrix Σ_k corresponding to various models: (1) Equal volume spherical clusters: $\Sigma_k = \lambda I$, where I is the identity matrix; (2) Spherical clusters of possibly unequal volume: $\Sigma_k = \lambda_k I$; (3) Elliptical clusters having equal volume, shape and orientation: $\Sigma_k = \lambda DAD^T$, where A is a diagonal matrix and D is an orthogonal matrix; (4) Unconstrained model: not imposing any structure on Σ_k.

The advantage of model-based clustering is that we do not need to heuristically judge which clustering result is the best which has to be done with most other clustering procedures. One could fit the model with different values of r and different structures of Σ_k and then pick up a best model according to a specific criterion function such as AIC and BIC.

Target estimation and stochastic approximation can be combined to well correct the biase of Pearson's correlation coefficients of genes. The details of this aproach and good performance are presented in chapter 2, section 2.4.3 so we won't talk much about it in this chapter. Here we apply it to estimate the correlation matrix of genes in the same cluster.

4.2.2 Statistical Model and Procedures

Our statistical model:

$X_{gij} = \mu_{gj} + \sigma_{gj}\epsilon_{gij}$

where X_{gij} is log transformed and suitably normalized intensities; μ_{gj} is the mean intensity of the gth gene in the jth group; σ^2_{gj} is the variance of the gth gene; and g(g=1,...G) indexes the genes on the microarray; j(j=1,2) indexes the two groups and i(i=1,...,n_i) indexes the objects. Generally, we consider a balanced design where $n = n_1 = n_2$.

Let $\underline{x}^1 = (x_1^1, x_2^1, \ldots, x_g^1, \ldots, x_G^1)$ and $\underline{x}^2 = (x_1^2, x_2^2, \ldots, x_g^2, \ldots, x_G^2)$ denote the random vectors of gene expressions in control group and treatment group respectively. We assume $\underline{x}^1 \sim MVN(\underline{\mu}^1, \Sigma^1)$ and $\underline{x}^2 \sim MVN(\underline{\mu}^2, \Sigma^2)$

Procedures:

(B1) Do a model based clustering analysis. After this step, suppose we have C clusters and the number of genes in cluster k is G_k.

(B2) Assign an error rate $\alpha_k = \alpha G_k/G$ to cluster k. In general, $\alpha = 0.05$

(B3) For each cluster k, calculate different powers $\beta_k(n_1, n_2, \delta)$ based on different sample sizes (n_1, n_2) and effect sizes (δ) controlling family wise error α_k. The calculation of power $\beta_k(n_1, n_2, \delta)$ for cluster k contains 7 substeps:

(B3.1) Use S.A.D. to estimate the correlation matrix of two groups $\hat{\Sigma}_1$ and $\hat{\Sigma}_2$.

(B3.2) Determine cutoff p-values for up-regulated genes and down-regulated genes p_{upcut} and $p_{downcut}$ (More details are provided after (B4)).

(B3.3) Sample control group data matrix from $MVN(0, \hat{\Sigma}_1)$ with sample size n_1^*.

(B3.4) Sample treatment group data matrix from $MVN(\hat{\delta}, \hat{\Sigma}_2)$ with sample size n_2^*, where $\hat{\delta}$ is an effect size vector and its elements are either δ, $-\delta$ or 0.

(B3.5) For each gene, calculate the probability p=Pr(t¿observed value), where $t \sim t_{n_1+n_2-2}$. If $p < p_{upcut}$ or $p > p_{downcut}$, then this gene is considered to be differentially expressed.

(B3.6) Calculate the percentage of differentially expressed genes detected.

(B3.7) Repeat (B3.1)-(B3.6) 1000 times and get a mean power.

(B4) Let the proportion of differentially expressed genes in cluster k is π_0^k. The overall power $\beta(n_1, n_2, \delta) = \frac{\beta_1 \pi_0^1 G_1 + \beta_2 \pi_0^2 G_2 + \cdots + \beta_C \pi_0^C G_C}{\pi_0^1 G_1 + \pi_0^2 G_2 + \cdots + \pi_0^C G_C}$

(B5) For every pair (n_1^*, n_2^*), we have an overall power. Smallest pair that achieve the desired power is selected as sample size.

To determine cutoff p-values for up-regulated genes and down-regulated genes controlling family wise error, we estimate the distribution of t statistic p-values under null hypothesis and choose the $1-\alpha_k/2$ quantile of maximum and $\alpha_k/2$ quantile of minimum as p_{upcut} and $p_{downcut}$. The detailed process is as follows:

(C1) Sample nn (say 20) objects from $MVN(0,\hat{\Sigma}_1)$ as control group and sample nn objects from $MVN(0,\hat{\Sigma}_2)$ as treatment group.

(C2) Calculate the probability Pr(T > sign$*$two sample t statistic) for each gene where $T \sim t_{2n-2}$.

(C3) Let $p_{(1)}$ be the minimum and $p_{(G)}$ be the maximum of p-values from (C2).

(C4) Repeat (C1)- (C3) m (for example 1000) times and get $(p^1_{(1)}, p^2_{(1)}, ..., p^m_{(1)})$ and $(p^1_{(G)}, p^2_{(G)}, ..., p^m_{(G)})$. Then we use $Q_{(\alpha_k/2)}$ of $p_{(1)}{}'s$ as cutoff p-value for up-regulated genes and $Q_{(1-\alpha_k/2)}$ of $p_{(G)}{}'s$ as cutoff p-value for down-regulated genes.

Why family wise error controlled in our methodology? Here we will prove it in the situation that there is only on cluster.

By the definition, the family wise error is $P($ at least one gene is considered diffentially expressed $\mid H_0)$

$=P(p_i < p_{upcut}$ or $p_i > p_{downcut}$ for some $i = 1, \cdots, G|H_0)$

$=P(p_{(1)} < p_{upcut}$ or $p_{(G)} > p_{downcut}|H_0)$

$=1 - P(p_{upcut} < p_{(1)} < p_{(G)} < p_{downcut}|H_0)$

However, $p_{upcut} \to Q_{0.025}$ of $p_{(1)}$, $p_{downcut} \to Q_{0.975}$ of $p_{(G)}$

Hence, family wise error $= 1 - P(p_{upcut} < p_{(1)} < p_{(G)} < p_{downcut}|H_0) < 0.05$

Two extreme cases: when genes are all independent, namely, the correlation matrix of genes are $\boldsymbol{I}$, our approach gives us the close results to t test based approach using bonferonni correction because

$p^i_1, p^i_2, \cdots, p^i_G$ i.i.d $U[0,1]$, so $p^i_{(1)}$ $Beta(1,G)$.

Given $p^1_{(1)}, p^2_{(1)}, \cdots, p^{1000}_{(1)}$ i.i.d $Beta(1,G)$, $Q_{(0.025)} \approx 1 - 0.975^{\frac{1}{G}}$

$P(p_i < Q_{(0.025)}$ for some $i = 1, 2, \cdots, G|H_0)$

$=1 - P(p_i > Q_{(0.025)}, i = 1, 2, \cdots, G|H_0)$

=1-0.975=0.025

And when genes are all pair wise linear correlated, $p_1^i = p_2^i = \cdots = p_G^i = p_{(1)}^i = p$,

Given $p_{(1)}^1, p_{(1)}^2, \cdots, p_{(1)}^{1000}$ i.i.d $U(0,1)$, $Q_{(0.025)} \approx 0.025$

$P(p_i < Q_{(0.025)}$ for some $i = 1, 2, \cdots, G|H_0)$

$=P(p < Q_{(0.025)}|H_0)$

=0.025 which is consistent with the results of testing one gene only.

4.3 Simulation

We have done three parts of simulation study. First, we evaluate the performance of our approach on highly correalted genes set. Since the whole set of genes are highly correlated in these simulated data we do not need to do clustering analysis. In this case, we only have one cluster, i.e. $C = 1$, $G_1 = G$ and $\alpha_1 = \alpha$ in step (B1) and (B2). In the second part, we check the perfrmance of model-based clustering on simulated data. Last, we applied our methodologies to simulated data and real data which shows clustered pattern.

Part I: We start simulating a dataset with 100 genes and 4 samples each group from $MVN(0, \Sigma)$ where Σ is a 100×100 positive finite matrix. The average absolute value of correlation matrix is 0.8. We applied two methods to this smulated data set.

Method 1 is our new method. In this case, image plot (See figure 4.1) shows data are highly correlated in one cluster. We calculate the mean power of different sample sizes controlling family wise error.

Method 2 is two sample t tests based approach using bonferonni correction. The procdures are as follows:

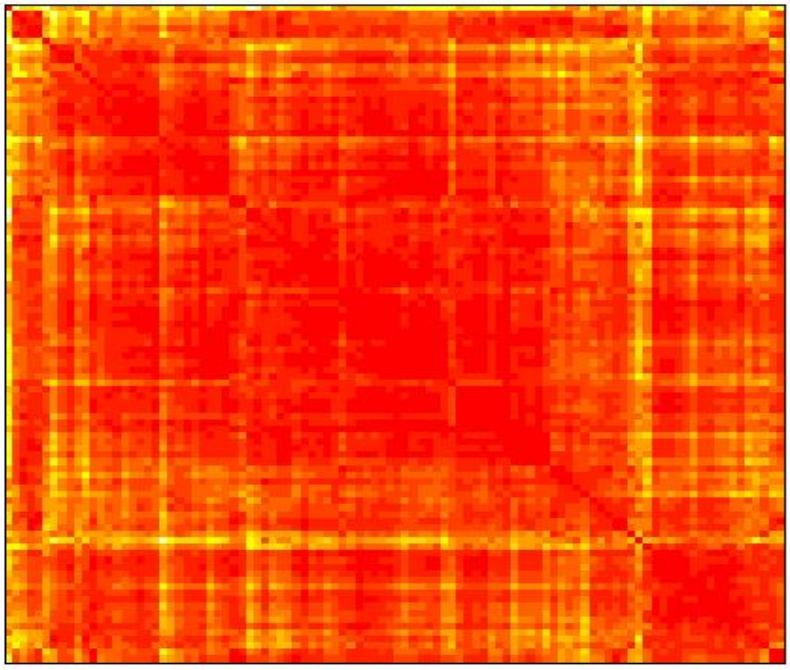

Figure 4.1: Simulated 100 genes data set: Ordered Gene Distance Matrix. Red represents small distance and white represents large distance.

S1: Sample control group data matrix (100*n) independently from standard normal distribution, e.g $N(0,1)iid$

S2: Sample treatment group data matrix (100*n) independently from normal distribution $N(\Delta,1)$, where Δ is desired detectable fold change.

S3: Two sample t-test used and cutoff p-value 0.05/100 used for each gene

S4: Calculate the percentage of differentially expressed genes detected

Repeat S1-S4 1000 times to get a mean power

Figure 4.2 plots power estimation vs. sample size calculated by two methods when effect size is 3, 2, 1.5 and 1 based on simulated 100 genes data set. From Figure 4.2, it is easily seen that our approach is much better than t test based approach in all four situations. For example, in the lower right graph, where the desired fold change is 1, 30 samples each group are required by t test based approach while only 17 samples are required by our approach.

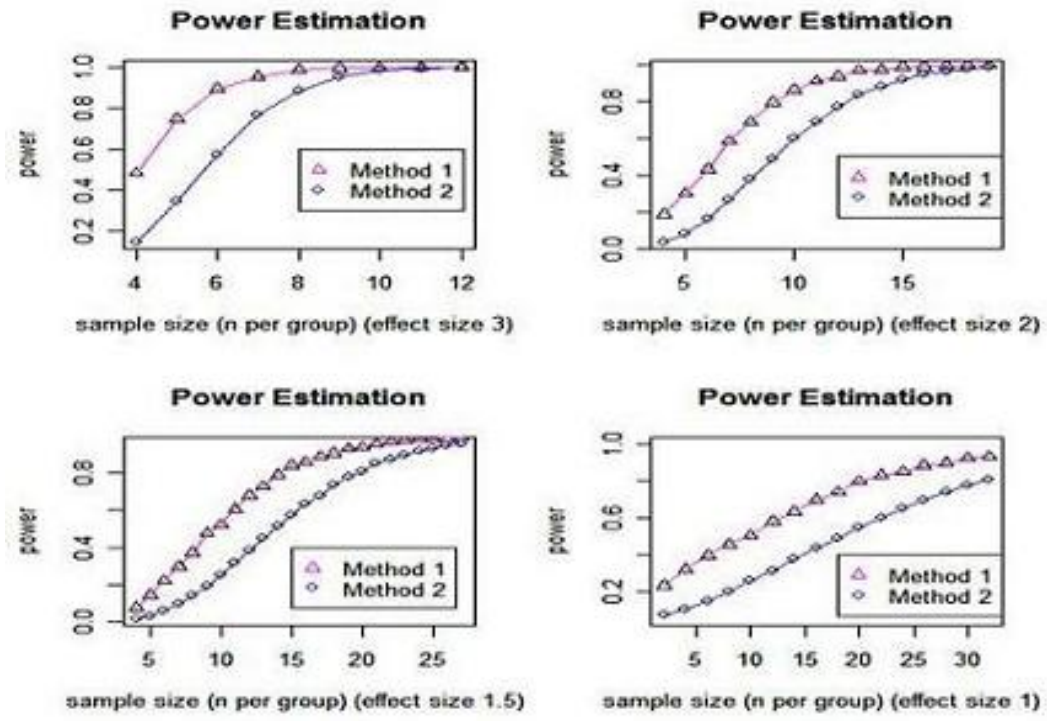

Figure 4.2: power estimation vs. sample size calculated by two methods when effect size is 3, 2, 1.5 and 1 based on simulated 100 genes data set

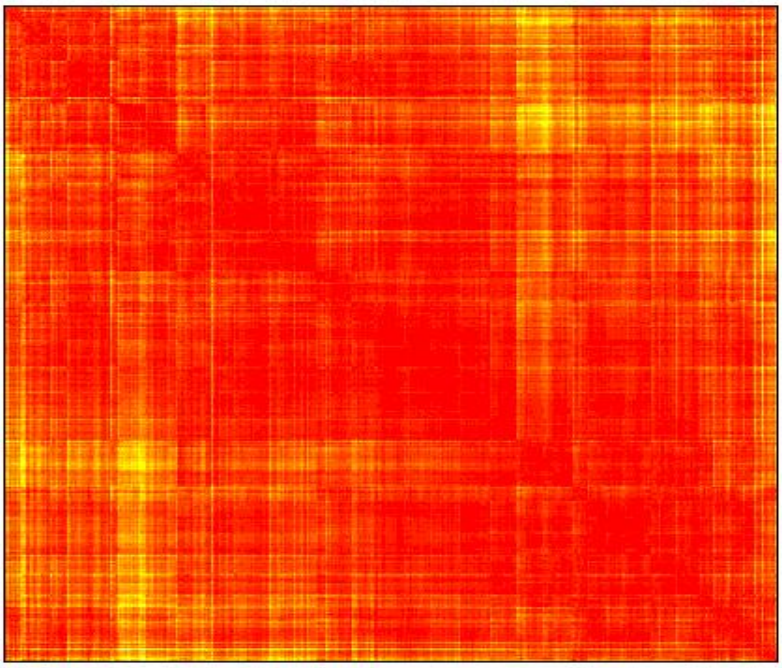

Figure 4.3: Simulated 100 genes data set: Ordered Gene Distance Matrix. Red represents small distance and white represents large distance.

Another dataset we simulate contains 1000 genes and 20 samples each group from $MVN(0, \Sigma)$ where Σ is a given positive finite matrix. Same as simulation I, two methods are used and the comparison is made. We consider 25 percent of genes are significantly differentially expressed. Results of power estimation and sample size calculation when effect size is 3, 2, 1.5 and 1 are shown in Figure4.2.

Part II: In our procedure, we choose model-based clustering method proposed by Yeung et al.(2001) because of the follwing two reasons:

(1) The distribution they assume for each cluster is multivariate normal distribution which is consistent with our assumption.

(2) Computing is easy since the methodology has already been implemented in R package mclust.

In their original paper, datasets used to demonstrate the performance of clustering methods are of small number of genes and do not fit our scheme. So we

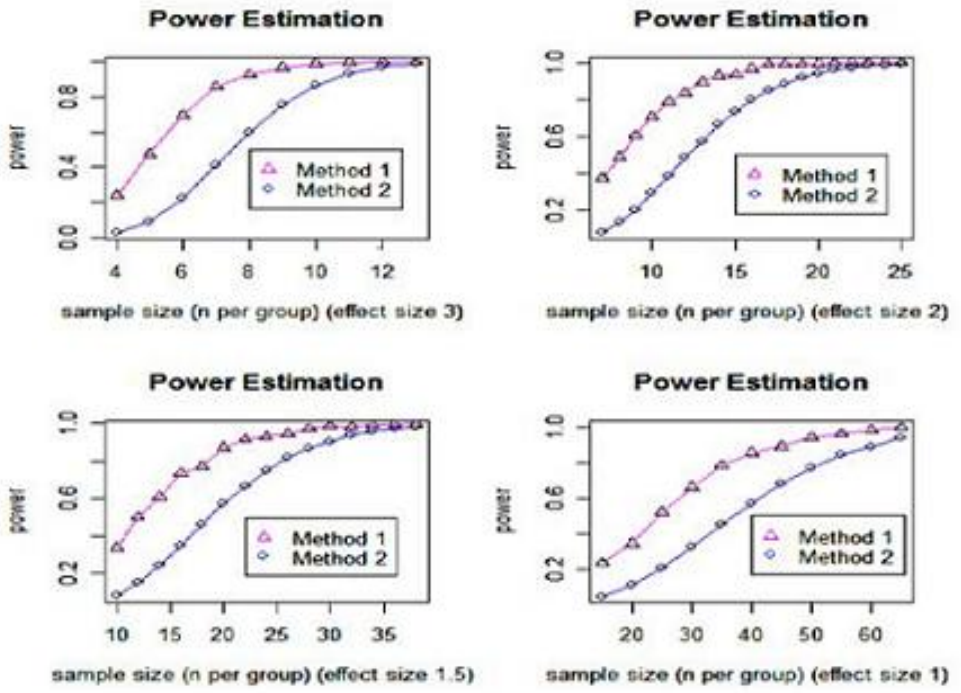

Figure 4.4: power estimation vs. sample size calculated by two methods when effect size is 3, 2, 1.5 and 1 based on simulated 1000 genes data set

simulate some datasets to check how it performs in our settings.

Part III: As we state before, when the whole set of genes are highly correlated, we do not need do clustering analysis. But when data displays clustered pattern, we need make clustering analysis to improve power. Let's see an example: our data set contains 500 genes in which 100 genes are pairwise linear correlated and the other 400 are independent. If we use t test based approach, the critical t value for each gene t_1^* satisfies $p(|t| > t_1^*) \approx 0.05/500$. And if we use ur approach and do not cluster the data, the critical t value for each gene t_2^* satisfies $p(|t| > t_2^*) \approx 0.05/401$. Since the two critical values are close, the statistical power won't be improved much!

We simulate a dataset with 500 genes in which 100 highly correlated genes are independent of the other 400 highly correlated genes. Model based clustering approach is used and data is split perfectly. We are interested in the 400 genes and assume the 400 genes are differentially expressed. Results of power estimation and sample size calculation when effect size is 3 or 1.5 are shown in Figure4.6.

4.4 Discussions

We check the possible cluster pattern among genes before power calculation in our methods. Acctually, if this step is skipped, the performance is still pretty good. But when there are too many genes, say 50,000, clustering is strongly recommended to screen genes and select interested clusters because S.A.D will cost lots of time in estimating the big N*N correlation matrix and if N is too big, general personal computers even dont have enough memory to support this estimation.

The sample size determined by our approach rely upon the association among genes. The stronger association, the less sample size which coincides common sense. This method is not recommended if the genes are independent or very

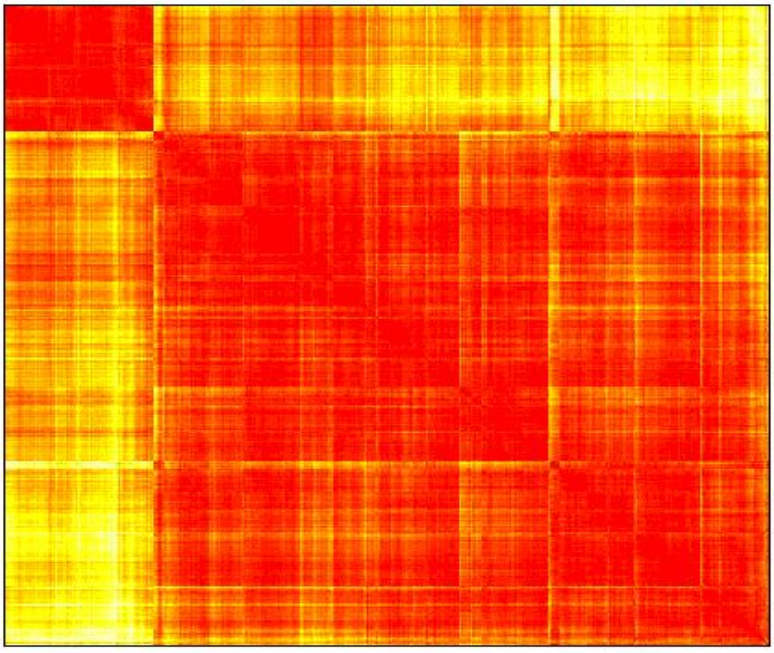

Figure 4.5: Simulated 500 genes data set: Ordered Gene Distance Matrix. Red represents small distance and white represents large distance.

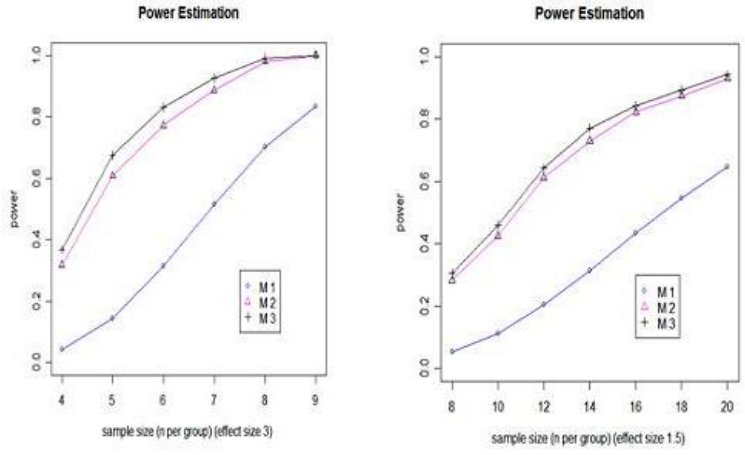

Figure 4.6: Power Estimation by three methods based on simulated 500 genes data Black: New approach with clustering; Pink:New approach without clustering; Blue: t test approach

weakly associated since our method will give us the similar result of t test approached using bonferroni adjustment which is very consertve.

In our approach, we used standard t test to calculate the statistics. T test can be replaced by Conditional t test or improved Conditional t test, the performance will be assessed in future work.

Chapter 5
Analysis of Gene Co-Expression Network

Abstract Graph based approaches are increasingly used to explore the functionality of genes. Gene co-expression networks are one of the examples. The concept is straightforward: nodes represent genes and nodes are connected if the corresponding gene pairs are significantly co-expressed. In this thesis, we build up a weighted gene co-expression network by converting the co-expression measure into a connection weight. We apply our method to simulated data and to a real microarray example and compared our method to other methods.

5.1 Introduction

Networks are defined by a series of points (nodes) interconnected by communication paths. Networks are increasingly used in biology and genetics because they provide an effective way to summary genes and proteins correlations. Types of networks include protein interaction networks (Uetz et al, 2000; Ito et al, 2001; Jeong et al, 2001; Wagner, 2001), metabolic networks (Fell and Wagner (2000); Jeong et al.(2000); Ma and Zeng (2003)), gene co-expression networks (Snel et al. (2002)) etc. In this chapter, we focus on gene co-expression networks, in which nodes represnt genes and nodes are conneted if the correponding gene pairs are significantly co-expressed.

Gene co-expression networks provide the interaction between individual genes and a system-level view of the organism and are widely used to explore the functioning of a cell.

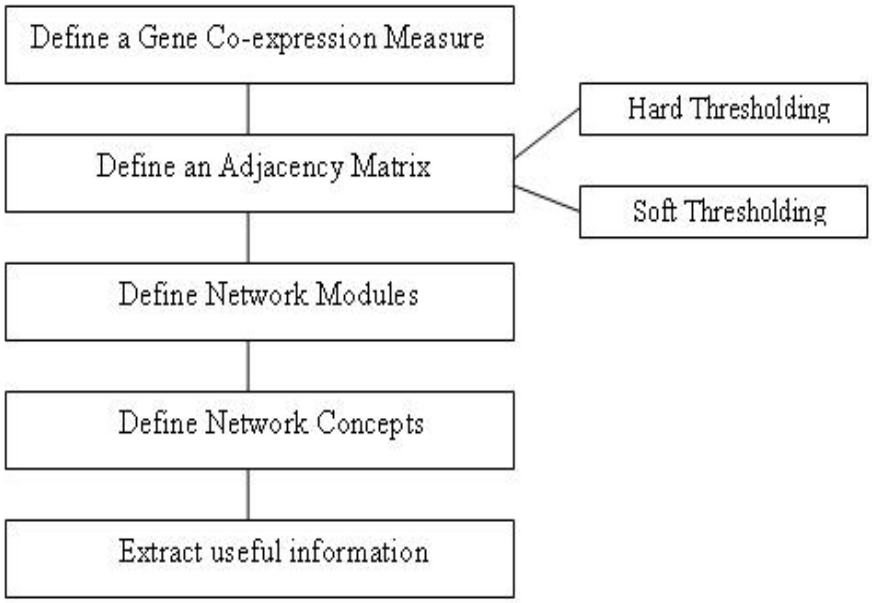

Figure 5.1: Flowchart of building a gene co-expression network

Below, we describe a general way to build gene co-expression networks. In Section 5.2, we present the steps of the network analysis. Then we talked some network concepts in Section 5.3. At last, we use simulated data and real data to compare our methods to other methods in Section 5.4. Conclusions are summarized in section 5.5.

5.2 Steps of Gene Co-expression Network Analysis

A flowchart of build a gene co-expression network is shown in Figure 4.1. This is tailored from Zhang and Horvath(2005)'s to fit our situations.

5.2.1 Define a Gene Co-expression Measure

A gene co-expression measure is needed to measure the level of concordance between gene expression profiles (Zhang and Horvath (2005)). Let m_{ij} denote the

co-expression measure of gene i and gene j, the most widely used one is the absolute value of Pearson correlation coefficient, i.e. $m_{ij} = |cor(i,j)|$. When samples of genes are small, which is normal in gene expression data, Pearson correlation coefficients are greatly biased and target estimation could be used to correct the biase and reduce the mean square error of Pearson correlation coefficient. The algorithm of target estimation used in correlation matrix estimation is presented in Chapter 2 thus we do not show any details here. We will use the absolute value of corrected Pearson correlation coefficient as gene co-expression measure, namely, $m_{ij} = |cor^*(i,j)|$.

5.2.2 Define an Adjacency Matrix

Each network corresponds to an adjacency matrix which encodes the connection strength between each pairs of nodes. The adjacency matrix can be obtained by thresholding the gene co-expression measure matrix $M = (m_{ij})$ (Butte and Kohane (2000); Carter et al. (2004); Davidson et al.(2001)). There are two ways to pick a threshold: one way is picking a 'hard' threshold (a number) based on the notion of statistical significane so gene co-expression is encoded using binary infromation(conneted=1, unconnected=0). Let $A = (a_{ij})$ denote the adjacency function, then the transformation from measure matrix can be described by a signum function:

$$a_{ij} = signum(m_{ij}) = \begin{cases} 1, & m_{ij} \geq \tau \\ 0, & m_{ij} < \tau \end{cases}$$

The drawbacks of 'hard' thresholding include loss of information of the magnitude of gene connections and sensitivity to the choice of the threshold(Carter et al.,2004). Moreover, an important question is whether it is biologically meaningful.

the other way is called 'soft' thresholding which weighs each connetion by a number between 0 and 1.

'Hard' threshodling results in unweighted networks while 'soft' thresholding results in weighted networks.

5.2.3 Define Network Modules

Ravasz et al.(2003) define modules as groups of nodes with high topological overlap. Different from that, our definition, adopted from Bergmann et al.(2004), is that modules are groups of genes whose expression profiles are highly correlated across samples. In general, hierarchical clustering method will be performed to generate a clustering tree and genes modules correspond to the branches of the tree(dendrogram). It is the simplest way that choosing a height cutoff to cut branches off the tree although it is not necessarily the best way. The choice of height cutoff is kind of arbitrary as in all hierachical clustering analyses. Ususally we pick up a balance point between the number of clusters and properties of dendrogram.

5.2.4 Define Network Concepts

Once the network has been constructed, one can explore the relationship among network concepts. One could study the properties of each cluster such as connectivity strength and number and percentage of connectivies or strong connectivies. Also, the relationship among clusters can be assessed such as calculating the correlation among modules and so on and compare different modules. It suggests to combine two clusters if the corresponding module genes are highly correlated. Modules showing similar network should have similar properties.

5.2.5 Extract useful information

The main usage of gene co-expression network is to extract useful biological nformation. From the constructed network, we could explore the functionality and

pathway of genes, identify essential genes susceptible to diseases etc. For example, we will pay special attention to the gene which carry the strongest connectivity in a module because it could be a crucial gene to predict functionality or detect a disease. And the genes with strong association with this gene are of interest too.

There are a few post-genomic methods which are used to analyze gene co-expression networks. However, it is beyond the scope of this thesis.

5.2.6 Comparison with Zhang and Horvarth (2005)

The comparison of our approach with Zhang snf Horvarth (2005)'s is shown in Table 5.1. The differences of two methods lie in two points: One point is different adjaccency matrix; the oher point is different dissimilarity measure. They defined the adjacency matrix as the power function or sigmoid function of the absolute value of Pearson's correlation coefficient and they defined the dissimilarity measure as the topological overlap matrix substraced fom one. Our adjacency matrix is the absolute value of estimated genes correlation matrix by S.A.D and the dissimilarity measure is correlation based distance.

5.2.7 A simulated example

We used the example mentioned in section 2.4. Data comes from multivariate normal with mean 0 and variance C. First we estimate the correlation matrix by target estimaton and stochastic approximation. Then the absolute value of estimated correlation was chosen as co-expression similarity measure and adjacency matrix. For Module detection, we conduct average linkage hierarchical clustering coupled with the dissimilarity measure $d_{ij} = \sqrt{1 - a_{ij}^2}$. To compare the performance of two methods, we tried different height cut-off values and then calculated corresponding missclasification rates. The smallest misclarification rate our approach can achieve is 117 when height cut-off value is 0.94 while their method can

Table 5.1: Comparison of Two Network Construction Methods

	Zhang and Horvath(2005)	*New*
Gene Co-expression Measure	$m_{ij} = \|cor(i,j)\|$ Pearson's Correlation	$m_{ij} = \|cor^*(i,j)\|$ Estimated Correlation
Adjacency Matrix	$a_{ij} = power(m_{ij})$ or $a_{ij} = sigmoid(m_{ij})$	$a_{ij} = m_{ij}$
Dissimilarity Measure	$d_{ij} = 1 - \frac{l_{ij}+a_{ij}}{min\{k_i,k_j\}+1-a_{ij}}$ $l_{ij} = \sum_u a_{iu}a_{uj}$ and $k_i = \sum_u a_{iu}$	$d_{ij} = \sqrt{1-a_{ij}^2}$

approach 144 if height cut-off value is 0.88. Our method has better performance in the sense that it can achieve smaller misclasification rate.

5.2.8 Application to Yeast Cell-Cycle Microarray Data

Yeast cell-cycle micrarray data contains 44 samples and it recorded gene expression levels during different stages of the cell cycles in yeasts. The yeast data are described in Eisen et al.(1998). We chose a subset of genes of it omitting the genes with lots of missing values and balancing the essential genes and non-essential genes. The final dataset contains 1290 genes and 44 samples where 645 genes are essential for yeast survival and the other half genes are not. Figure 5.2 displays the color coded picture of correlation among genes.

Similarly, we used S.A.D to estimate the correlation matrix of genes, then follow the flowchart (Figure 5.1) to construct the gene network. The average linkage hierarchical clustering was undertaken where the dissimilarity measure $d_{ij} = \sqrt{1-a_{ij}^2}$. The clustering tree is shown in Figure 5.3(A). We choose .97 as the cut-off height and Figure 5.3(B) displays the corresponding branch colors. To examinine the gene essentiality, we plotted the essential genes in Figure 5.3(C). It is easily seen that essential genes are concentrated in yellow and turquoise module. Figure 5.4 plots the mean gene significance in modules and the 95% confidence interval of the mean which is consistent with Figure 5.3(C). These plots using R

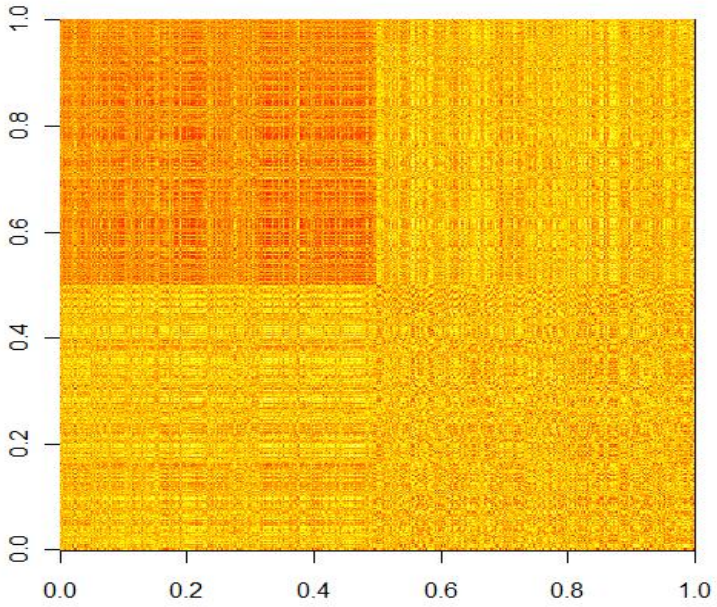

Figure 5.2: Distance Matrix of Yeast data

codes from http://www.genetics.ucla.edu/labs/horvath/GeneralFramework/.

5.3 Discussions

This chapter presents a procedure of building a gene co-expression network. The difference among network construction methods comes from different choices of gene co-expression measure, dissimilarity measure, adjacency Matrix, clustering methods and module selection. Our approach is very straightforward and it is very easy to understand. We emphasized on the good performance of our approach on general datasets and presents a common gene network construction methods here. While in the future, we will explore more about the properties of gene network and we will also do some research on special networks.

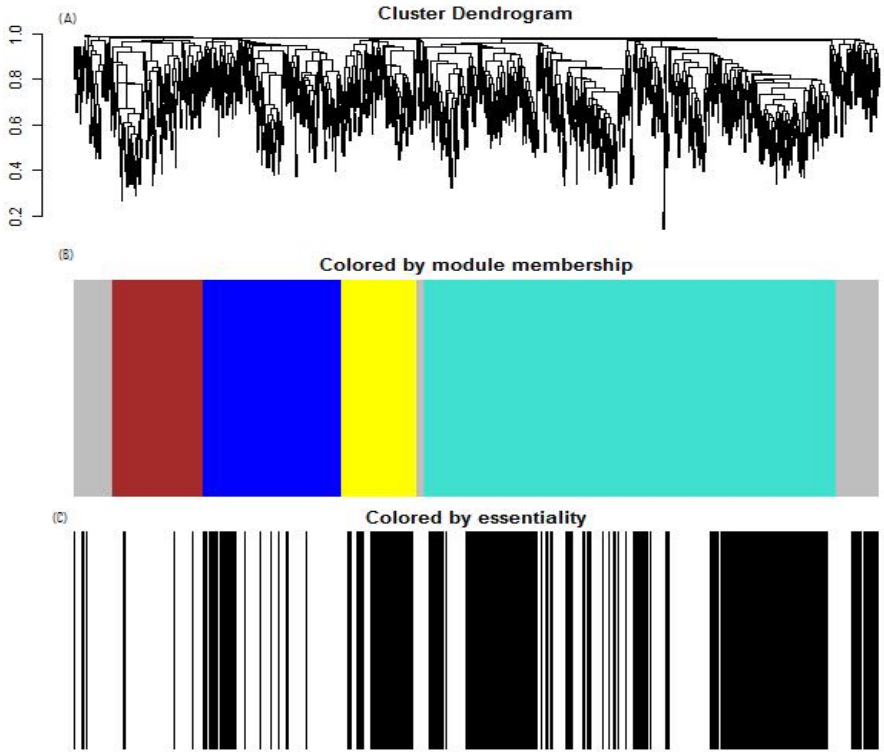

Figure 5.3: Results of Yeast Data (A):Clustering tree; (B):Corresponding branch colors; (C): Essential genes (black)

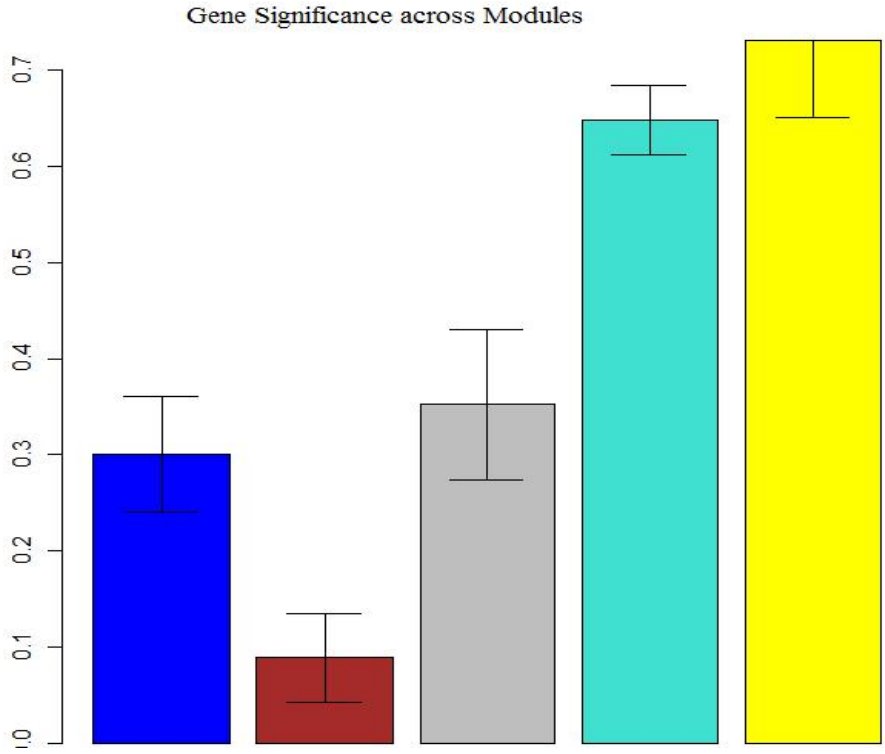

Figure 5.4: Gene Significance across Modules

Chapter 6
Conclusions and Future Work

In this thesis, we first talked about stoachstic approaximation for distributions (S.A.D.) which produces a series of distribution functions which converge to true distribution function in probablity under moderate conditions. One application of S.A.D to microarray data is to estimate the common distribution of standard deviation of all genes. With it, we could generate an envelope of t-values conditional on standard deviation which are aimed to judge whether a gene is significantly dfferentiately expressed or not. Comparison results on simulated dataset and real data set demonstrate that the proposed improved Conditional t test is better than Conditonal t test thus it is better than SAM and t test. S.A.D. can also be undertaken to estimate correlation matrix of genes which is a challenge for statisticians in microarray data analysis. It shows superiority to Pearson's correlation coefficients. Then A sample size determination method including model-based clustering and accurately estimated correlation matrix is presented and controlling family wise error was presented. It is much better than t test based approach. Last we described the steps of our gene network construction method and showed the simplicity and better performance than the method in the literature.

Although all these methods are aimed to do microarray data analysis, their usages are not limited to microarray data. They works well for any large dataset with few replicates.

In the future, we have a few things could do in our mind. First, we will try to improve the algoithm of S.A.D to make it more efficient and save user computing

time. when determine the sample size we need. Then when we determine the sample size needed, we could replace t test approach with Conditional t approach or improved Conditional t approach and compare the performance. In our paper, we only consider the cases that there are only two groups in microarray data. We will exploer more if there are two or more groups. Last, we are plan to do some research on special networks and explore more about perperties of gene co-expression network.

References

[1] Ash 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. Powel, L. Yang, G. E. Marti, T. Moore, J. Hudson, Jr, 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 February 3, 2000

[2] Amaratunga, D. and J. Cabrera. Conditional t. Unpublished manuscript, 2003

[3] Amaratunga, D. and J. Cabrera. Exploration and analysis of DNA microarray and protein array data. Wiley,2004

[4] Amaratunga, D. and J. Cabrera. A conditional t suite of tests for identifying differentially expressed genes in a DNA microarray experiment with little replication. Statistics in Biopharmaceutical Research, 1:26-38, 2007

[5] Asyali, M.H. and M. Alci, Reliability analysis of microarray data using fuzzy C-means and normal mixture modeling based classification methods. Bioinformatics. Vol. 21, No. 5, pp. 644-649, 2005.

[6] Baldi, P., and A. D. Long. A Bayesian framework for the annalysis of microarray expression data: Regularized t-test and statistical inferences of gene changes. Bioinform., 7,509-519, 2001.

[7] Benjamini, Y. and Y. Hochberg. Controlling the False Discovery Rate: A practical and powerful approach to multiple testing. J. Roy. stat. Soc., B57, 289-300, 1995.

[8] Broger. P. Ranking genes with respect to differential expression. Genome Biol.,3, preprint 00007. 1-preprint 0007.23, 2002.

[9] Bolstad, B. M., R. A. Irizarry, M. Astrand and T. P. Speed A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. Bioinformatics, 19, 185193, 2003.

[10] Cabrera, J. and L. T. Fernholz. Target estimation for bias and mean square reduction. Annals of Statistics. 27, 1080-1104, 1999.

[11] Cabrera, J. and P. Meer. Unbiased Estimation of Ellipses by Bootstrapping. IEEE Trans. Pattern Anal. Machine Intell. Vol. 18, 752-756, 1996

[12] Cabrera, J. and G. S. Watson. Simulation Methods For Mean and Median Bias Reduction. Statistical Planning and Inference, 57, 143-152, 1997

[13] Chambers, J., A. Angulo, D. Amaratunga, H. Guo, Y. Jiang, J. S. Wan, A. Bittner, K. Frueh, M. R. Jackson, P. A. Peterson, M. G. Erlander, and P. Ghazal. DNA microarrays of the complex human cytimegalovirus genomeL Profiling kinetic class with drug sensitivviral gene expression. J.Virol.,73,5757-5766,1999.

[14] Cheng, Y. and G. M. Church. Biclustering of expression data. Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology (ISMB) 8, 93103, 2000.

[15] Chipman, H. and R. Tibshirani. Hybrid hierarchical clustering with applications to microarray data. Biostatistics 7, 286301, 2006.

[16] Dembele, D. and P. Kastner. Fuzzy C-means method for clustering microarray data. Bioinformatics Vol. 19 no. 8, 973-980, 2003

[17] DeRisi, J. L., V. R. Iyer, and P. O. Brown. Expoloring the metabolic and genetic control of gene expression on a genomic scale. Science, 278,680-686, 1997.

[18] Dudoit, S. and J. Fridlyand. A prediction-based resampling method for estimating the number of clusters in a dataset. Genome Biology, 3(7),research0036.1research0036.21, 2002.

[19] Dudoit, S., Y. H. Yang, M. C. Callow, and T. P. Speed. Statistical methods for identifying differentially expressed genes in replicated cDNA microarray experiments. Stat. Sinica, 12, 111-140, 2002.

[20] Dobbin, K. and R. Simon (2007), Sample size planning for developing classifiers using high dimensional DNA microarray data, Biostatistics. 2007 Jan;8(1):101-17.

[21] Durbin, B., J. Hardin, D. M. Hawkins and D. M. Rocke. A variance-stabilizing transformation for gene-expression microarray data. Bioinformatics, 18, 105S110S, 2002.

[22] Eisen, M.B., P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-eide expression patterns. Proc. Nat. Acad. Sci., 95, 14863-14868, 1998.

[23] Friedman, J. H. and J. J. Meulman, Clustering objects on subsets of attributes. Unpublished manuscript,2002.

[24] Golub, T.R., 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, 531-537, 1999.

[25] Higham, D. J., G. Kalna And M. Kibble. Spectral clustering and its use in bioinformatics. Journal of Computational and Applied Mathematics 204, 2537, 2007.

[26] Iyer, V. R., M. B. Eisen, D. T. Ross, G. Schuler, T. Moore, J. C. F. Lee, J. M. Trent, L. M. Staudt, D. Shalon, D. Botstein and P. O. Brown. The transcriptional program in the response of human fibroblasts to serum. Science, 283,83-87, 1999.

[27] Khan, J., 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.

[28] Laan, M. V. and K. Pollard. A new algorithm for hybrid hierarchical clustering with visualization and the bootstrap. Journal of Statistical Planning and Inference, 117:275303, 2003.

[29] Li, S. S., J. Bigler, J. W. Lampe1, J. D. Potter and Z. Feng. FDR-controlling testing procedures and sample size determination for microarrays, Statist. Med. 24:2267-2280, 2005;

[30] Liu, P. and J. T. Gene Hwang (2007) Quick calculation for sample size while controlling false discovery rate with application to microarray analysis, Bioinformatics, Vol. 23 no. 6 739-746, 2007

[31] Madeira, S. C. and A. L. Oliveira. Biclustering algorithms for biological data analysis: a survey. IEEE Transactions on Computational Biology and Bioinformatics 1, 2445, 2004.

[32] McLachlan, G. J., R. W. Bean, and D. Peel. A mixture model-based approach to clustering of microarray expression data. Bioinform.,18,413-422, 2002.

[33] Medvedovic, M. and S. Sivaganesan. Bayesian infinite mixture model based clustering of gene expression profiles. Bioinformatics 18: 1194-1206, 2002.

[34] Medvedovic, M., K. Y. Yeung and R. E. Bumgarner. Bayesian Mixtures for Clustering Replicated Microarray Data. Bioinformatics. 20: 1222-1232, 2004.

[35] Nowak, G. and R. Tibshirani. Complementary hierarchical clustering. Biostatistics, 9, 3, 467483, 2008

[36] Newton, M. A., C. M. Kendziorski, C. S. Richmond, F. R. Blattner and K. W. Tsui. On differential variability of expression ratios: Improving statistical inference about gene expression changes from microarray data. J.Comput. Biol., 8,37-52, 2001.

[37] Pan, W., J. Lin and C. Le. Model-based cluster analysis of microarray gene-expression data. Genome Biol., 3(2), research 0009.1-0009.8, 2002.

[38] Pan, W., J. Lin and C. Le. How many replicates of arrays are required to detect gene expression changes in microarray experiments? A mixture model approach. Genome Biology 3(5): research0022.1-0022.10, 2002.

[39] Pan, W. Incorporating gene functions as priors in model-based clustering of microarray gene expression data. Bioinformatics 22, 795801, 2006.

[40] Pawitan, Y., S. Michiels, S. Koscielny, A. Gusnanto and A. Ploner. False discovery rate, sensitivity and sample size for microarray studies. Bioinform., Vol. 21 no. 13, 30173024, 2005.

[41] Raychaudhuri, S., J. M. Stuart, and R. B. Altman. Princioal components analysis to summarize microarray experiments:Application to sporulation time series. Pacific Symp. Biocomputing,5,452-463, 2000.

[42] Robbins, H. and S. MoNRo. A STOCHASTIC APPROXIMATION METHOD. The Annals of Mathematical Statistics, Vol. 22, No. 3, 400-407, 1951.

[43] Schena, M., D. Shalon, R. W. Davis, and P. O. Brown. Quantitative monitoring of geneexpression patterns with a complementary DNA microarray. Science, 270,467-470, 1995.

[44] Shao, Y. and C. Tseng. Sample size calculation with dependence adjustment for FDR-control in microarray studies, Statist. Med. 2007

[45] Speed, T. Always log spot intensities and ratios. http://www.stat.berkeley.edu/users/terry/zarray/Html/log.html, 2001

[46] Storey, J. D. and R. Tibshirani Estimating false discovery rates under dependence, with applications to DNA microarrays. Technical Report 2001-18. Department of Statistics, Stanford University, Stanford, 2001.

[47] Storey, J. D. A direct approach to false discovery rates. J. R. Stat. Soc. B, 64, 479498, 2002.

[48] Tamayo, P., D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E. S. Lander, and T. R. Golub. Interpreting patterns of gene expression with selforganizing maps: Methods and application to hematopoietic differentiation. Proc. Nat. Acad. Sci., 96, 2907-2912, 1999.

[49] Toronen, P., M. Kolehmainen, G. Wong, and E. Castren . Analysis of gene expression data using selforganizing maps. FEBS Lett.,451,142-146, 1999.

[50] Troyanskaya, O. G., M. E. Garber, P. O. Brown, D. Botstein and R. B. Altman. Nonparametric methods for identifying differentially expressed genes in microarray data. Bioinformatics, Vol. 18 no. 11 2002 Pages 14541461, 2002

[51] Tsai,C., S. Wang, D. Chen and J. J. Chen. Sample size for gene expression microarray experiments. Bioinform., Vol. 21 no. 8, 15021508, 2005

[52] Tseng, G. C. Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data. Bioinformatics 23(17):2247-2255, 2007

[53] Tsodikov, A., A. Szabo and D. Jones. Adjustments and measures of differential expression for microarray data. Bioinformatics. v18. 251-260, 2002.

[54] Tusher, V. G., R. Tibshirani, and G. Chu. Significance analysis of microarrays applied to the ionizing radiation response. Proc. Nat. Acad. Sci., 98,5116-5121, 2001.

[55] Xiong, M., W. Li, J. Zhao, L. Jin and E. boerwinkle. Feature(gene) selection in gene expressio-based tumor classification. Mol. Genet. Metabol.,73,239-247, 2001.

[56] Yekutieli, D and Y. Benjamini Resampling-based false discovery rate controlling multiple test procedures for correlated test statistics. Journal of Statistical Planning and Inference, 82:171196, 1999.

[57] Yeung, K. Y., C. Fraley, A. E. Raftery and W. L. Ruzzo. Model-based clustering and data transformations for gene expression data. Bioinform., 17, 977-987, 2001.

[58] Yuan, M. and C.Kendziorski, A Unified Approach for Simultaneous Gene Clustering and Differential Expression Identification, Biometrics, 62(4), 1089-1098,2006

[59] Zien, A., J. Fluck, R. Zimmer and T. Lengauer. Microarrays: How many do you need? J. Comput. Biol., 10, 653667, 2003.

[60] Zhang, B and S. Horvath A General Framework for Weighted Gene Co-Expression Network Analysis, Statistical Applications in Genetics and Molecular Biology: Vol. 4: No. 1, Article 17, 2005.

Vita

Zhaoyu Luo

1997-2002	Attended Special Class for the Gifted Young, University of Science and Technology of China
2002	B.S in Mathematics, University of Science and Technology of China
2002-2007	Attended Department of Statistics and Biostatistics, Rutgers University
2003	Research Assistant, Department of Statistics and Biostatistics, Rutgers University
2003-2006	Teaching Assitant, Department of Statistics and Biostatistics, Rutgers University
2006	Statistician, Summer Intern, Merck Reaserach Lab, Rahway, New Jersey
2006-2007	Statistician, Merck Reaserach Lab, Rahway, New Jersey
2007-2008	Statistician, Sanofi-Aventis, Bridgewater, New Jersey
2008-2009	Statistician, Clinical Trials and Surveys Corporation, Baltimore, Maryland
2009	Ph.D in Statistics, Rutgers University