ABSTRACT

LIU, JIAJUN. Domain Enhanced Analysis of Microarray Data Using GO Annotations. (Under the direction of Dr. Jacqueline M. Hughes-Oliver and Dr. Jason Osborne.)

New biological systems technologies give scientists the ability to measure thousands of bio-molecules including genes, proteins, lipids and metabolites. We use domain knowledge, e.g., the Gene Ontology, to guide analysis of such data. By focusing on domain-aggregated results at, say the molecular function level, increased interpretability is available to biological scientists beyond what is possible if results are presented at the gene level. We use a “top-down” approach to perform domain aggregation by first combining gene expressions before testing for differentially expressed patterns. This is in contrast to the more standard “bottom-up” approach where genes are first tested individually then aggregated by domain knowledge. The benefits are greater sensitivity for detecting signals.

In DEA procedure, the first scores from the PLS procedure are used to test for differentially expressed patterns using the t test. We find the general t test inadequate for adjusting for the number of genes within each GO term. New tests are proposed by finding the true null distribution of each PLS score adjusted for the size of the GO term. Our method is assessed using a series of simulation studies. Furthermore, we also discuss the impact of our testing procedure with different coding of our classification response variable, namely 0/1 or -1/1 for data with two classes.
Domain Enhanced Analysis of Microarray Data Using the Gene Ontology Annotations

by

JIAJUN LIU

A dissertation submitted to the Graduate Faculty of North Carolina State University in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

STATISTICS
Raleigh, North Carolina
2007

Approved by:

Jacqueline M. Hughes-Oliver Chair of Advisory Committee
Jason Osborne Co-Chair of Advisory Committee

Sidney Stanley Young
Helen Zhang

Jack Alan Menius Geoff Benson
Biography

Jiajun Liu was born in Shanghai, China on Oct. 23, 1979. He received his Bachelor of Arts degree in Mathematics and Computer Science from Berea College in 2002 and his Master’s degree in Statistics from North Carolina State University in 2004. He is a member of Phi Kappa Phi Honor Society and Mu Sigma Rho, National Statistical Honor Society.
## Contents

List of Tables vi

List of Figures ix

1 Summary 1

2 Literature Review 9

2.1 Introduction of Microarray Technology .................................. 9

2.2 Tests for Differentially Expressed Genes .............................. 14

2.2.1 Fold Change ...................................................... 15

2.2.2 T-test .......................................................... 16

2.2.3 Modified t-test .................................................. 18

2.2.4 Mixture Model Approach ......................................... 23

2.2.5 Optimal Discovery Procedure ...................................... 30

2.2.6 Summaries ......................................................... 33

2.3 Adjustment for Multiple hypotheses Testing .......................... 36

2.3.1 Controlling the Family-Wise Error Rate .......................... 38

2.3.2 False Discovery Rate .............................................. 39

2.3.3 Positive False Discovery Rate .................................. 41

2.3.4 Local False Discovery Rate ..................................... 44

2.4 Gene Annotation Enhanced Analysis .................................... 47

2.4.1 Fisher’s Exact Test ............................................... 48

2.4.2 Gene Set Enrichment Analysis ................................... 49

2.4.3 Parametric Analysis of Gene Set Enrichment .................... 52

2.4.4 Meta-Analyses ..................................................... 53

2.4.5 Empirical Bayesian Generalized Linear Model ................. 55
References                  172
Appendices                 188
List of Tables

2.1 Contingency table to show relations between observed and real hypotheses. ................................. 37
2.2 Contingency table classification of genes according to whether they are identified as differentially expressed (flagged) and whether they are part of the GO term. ................................. 49

3.1 Contingency table classification of genes according to whether they are identified as differentially expressed (flagged) and whether they are part of the GO term. ................................. 65
3.2 Gene counts within fully or partially important molecular functions. ......................................................... 73
3.3 BH-adjusted p-values of important GO terms from different methods. These results are from a single simulation replicate where $\delta = 0.3$. P-values for Fisher’s exact test equal 1 in every case. . 74
3.4 GO-level analysis results for ten most significant GO terms found by Fisher’s exact test in Chiaretti’s data. ................................. 83
3.5 GO-level analysis results for ten most significant GO terms found by DEA-PLS in Chiaretti’s data. ................................. 84
3.6 GO-level analysis results for top 10 most significant GO terms found by Fisher’s exact test in Golub’s data. ................................. 89
3.7 GO-level analysis results for the most significant GO terms found by DEA-PLS in Golub’s data. ................................. 90
3.8 The most significant probe sets in Golub’s data. ................................. 90
4.1 Comparing 5th percentile of simulated $W^m$ and $N(-1m, 3m)$. . 111
4.2 Power calculation by changing mean difference $\mu$ of all genes related to each GO. $H_0: \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a: \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generate the power is 20.

4.3 Power calculation by changing mean difference $\mu$ of only one of the genes related to each GO. $H_0: \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a: \mu_{01} - \mu_{11} = 2\mu$, and $\mu_{0j} = \mu_{1j} = 0$ for $j = 2, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generate the power is 20.

5.1 Power calculation by changing mean difference $\mu$ of all genes related to each GO with response variable $y = [1, -1]$. $H_0: \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a: \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20.

5.2 Power calculation by changing mean difference $\mu$ of only one genes related to each GO with response variable $y = [1, -1]$. $H_0: \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a: \mu_{01} - \mu_{11} = 2\mu$, and $\mu_{0j} = \mu_{1j} = 0$ for $j = 2, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20.

5.3 The comparison of the $p = 0.05$ cutoffs for different response variables. $\Delta S_1$ denotes the statistic for variable $y = [0, 1]$. $(\Delta S_2)^2$ denotes the statistic for variable $y = [1, -1]$.

5.4 Power calculation by changing mean difference $\mu$ of all genes related to each GO with response variable $y = [1, -1]$. $H_0: \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a: \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20. There is not correlation between genes.
Power calculation by changing mean difference $\mu$ of all genes related to each GO with response variable $y = [1, -1]$. $H_0 : \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a : \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20. There is correlation between genes, pairwise correlation $\rho = 0.5$. 169
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>An illustration of the GSEA procedure.</td>
<td>51</td>
</tr>
<tr>
<td>2.2</td>
<td>An illustration of the permutation test of gene set enrichment analysis.</td>
<td>52</td>
</tr>
<tr>
<td>3.1</td>
<td>A flow chart illustrates our procedure. A: steps 1 and 2 aggregate genes and create latent variables. For PLS procedure, treatments are considered as response variable. B: steps 3 and 4 test for significance of each latent variable with respect to treatment and adjust p-values for multiple testing.</td>
<td>68</td>
</tr>
<tr>
<td>3.2</td>
<td>Sensitivity and specificity of four methods as a function of $\delta$ in the simulation study. Sensitivities are based on averages across all simulation replicates, with pointwise approximate 95% confidence intervals shown as vertical bars. Fisher’s exact test is solid line, DEA-PLS is dashed line, DEA-Mean is dotted line and GSEA is dashed-dotted line. $\alpha_I = 0.05$ for Fisher’s exact test. All GO-level p-values were BH-adjusted using $\alpha_G = 0.05$.</td>
<td>76</td>
</tr>
<tr>
<td>3.3</td>
<td>ROC curve for Fisher’s exact test by changing the threshold $\alpha_I$ for gene-level analysis. Data was generated following the simulation design using $\delta = 0.3$. Estimates are based on 50 simulation replicates. Vertical bars are pointwise approximate 95% confidence interval. Also shown are points for DEA-PLS, DEA-Mean, and GSEA; These are single points because they do not depend on $\alpha_I$.</td>
<td>78</td>
</tr>
</tbody>
</table>
3.4 GO graph of findings by Fisher’s exact test for Chiaretti’s Data. Specific terms are at the top, while general terms are at the bottom. Top ten significant GO terms are underlined, with rank shown in upper left of rectangle. Ranks for top ten probe sets are shown in curly brackets. ................................. 86

3.5 GO graph of findings by DEA-PLS for Chiaretti’s Data. Specific terms are at the top, while general terms are at the bottom. Top ten significant GO terms are underlined, with rank shown in upper left of rectangle. Ranks for top ten probe sets are shown in curly brackets. ................................. 87

4.1 Theoretical density plot of $W_1$ .............................................. 105
4.2 The theoretical Density plot of $W^2$ ................................. 107
4.3 Empirical distribution of $W^m$ ................................. 108
4.4 5th percentile of empirical distribution of $Y_m = \frac{W^m - m \mu}{\sigma \sqrt{m}}$ displayed as dots and its standard normal approximation displayed as a line. ................................. 112
4.5 Approximation of $f(w^1)$ by skew-normal distribution: the histogram is the empirical distribution of $W^1$ and the curve is the skew-normal fit. The exact 5th percentile is -4.45, compared to -4.123 from the skew normal fit. ................................. 113
4.6 Approximation of $f(w^2)$ by skew-normal distribution: the histogram is the empirical distribution of $W^2$ and the curve is the skew-normal fit. The exact 5th percentile is -6.75, compared to -6.549 from the skew normal fit. ................................. 114
4.7 Approximation of $f(w^3)$ by skew-normal distribution: the histogram is the empirical distribution of $W^3$ and the curve is the skew-normal fit. The exact 5th percentile is -8.73, compared to -8.631 from the skew normal fit. ................................. 115
4.8 Approximation of $f(w^4)$ by skew-normal distribution: the histogram is the empirical distribution of $W^4$ and the curve is the skew-normal fit. The exact 5th percentile is -10.83, compared to -10.579 from the skew normal fit. ................................. 116
4.9 5th percentile of $Y_m$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 5th percentiles of the empirical distribution of $Y_m$ based on 20000 simulated data points. ................................. 124
4.10 1st percentile of $Y_m$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 5th percentiles of the empirical distribution of $Y_m$ based on 20000 simulated data points. .................................................. 125

4.11 Compare alternative distribution with skew-t null distribution as $\mu$ increases with different values of $m$. .................. 129

4.12 Sensitivity and specificity of four methods as a function of $\delta$ in the simulation study. Sensitivities are based on averages across all simulation replicates, with pointwise approximate 95% confidence intervals shown as vertical bars. Fisher’s exact test is solid line, DEA-PLS using t null is dashed line, DEA-Mean is dotted line, GSEA is dashed-dotted line and DEA-PLS using skew-t null is long-dashed line. $\alpha_f = 0.05$ for Fisher’s exact test. All GO-level p-values were BH-adjusted using $\alpha_G = 0.05$. .... 132

5.1 Estimation of 95th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 95th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points. ................. 157

5.2 Estimation of 99th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 99th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points. ................. 158

5.3 Sensitivity and specificity of four methods as a function of $\delta$ in the simulation study. Sensitivities are based on averages across all simulation replicates, with pointwise approximate 95% confidence intervals shown as vertical bars. Fisher’s exact test is solid line, DEA-PLS using t null is dashed line, DEA-Mean is dotted line, GSEA is dashed-dotted line PAGE is skew-t null is long-dashed line and the new derived test is short-long-dashed lines. $\alpha_f = 0.05$ for Fisher’s exact test. All GO-level p-values were BH-adjusted using $\alpha_G = 0.05$. ................. 161
5.4 Examples of Correlations of two genes related to a GO terms from an experimental data set. $x$ axis and $y$ axis are the expression levels for the two genes for a particular sample. $r$ is the correlation between the genes over all. $r_n$ is the correlation between the genes for normal samples and $r_d$ is the correlation between genes for diseased samples. The red dots correspond to the diseased samples and the green dots correspond to the non-diseased samples. 

5.5 95th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 95th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points. The genes are simulated with pairwise correlation coefficient of $\rho = 0.5$. 

5.6 99th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 99th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points. The genes are simulated with pairwise correlation coefficient of $\rho = 0.5$. 
Chapter 1

Summary

Advances in microarray technology have greatly enhanced gene expression studies. In these studies, thousands of genes are monitored and probed on only a few to tens of samples. Gene expression data present both great opportunities and challenges. Patterns of gene expression can be used to determine genes with similar behavior, suggest biomarkers for a specific disease, and propose targets for drug intervention.

Major efforts have been put into analyzing microarray data. Many different methods are becoming readily available. One issue existing in most of these methods is how to interpret the results when hundreds of genes are found to be important. We started our research with the goal to increase interpretability
of microarray data analysis in general. Fortunately, biologists have created a number of data bases to explain the functionalities of most of the genes, for example, the Gene Ontology (GO) by Ashburner et al. (2000). The focus of our research hence changed to how to utilize this type of biological knowledge base to enhance microarray data analysis with the goal to increase interpretability of the analysis results. During the early stage of our research, a lot of time was dedicated to understanding the structure of the Gene Ontology and creating the mappings between the Gene Ontology and the microarray data on hand.

At that time, very little research had been published on pathway analysis on microarray data. It was mostly the standard “bottom-up” approach where genes are first tested individually then aggregated to biological function (GO terms) by domain knowledge. Depending on either the number of important (meaning that a hypothesis test concluded significant differences in expression levels across groups) genes or the rank ordering of importance of the genes in each GO term, tests for significant GO terms have been proposed. This includes the earlier approaches such as Fisher’s exact test (Draghici et al., 2003a), Gene Set Enrichment Analysis (GSEA) (Subramanian et al., 2005; Mootha et al., 2003), Parametric Analysis of Gene Set Enrichment (PAGE) (Kim and Volsky, 2005). Generally speaking, these “bottom-up” gene set enrichment methods improve the ability to interpret results from gene level analysis. The disadvan-
tage is that they all require gene-level analysis prior to conducting a GO-level analysis. The qualities of the results from the gene-level analysis are directly related to the effectiveness of these methods. For some of these procedures, we need to identify important or non-important genes, meaning that a statistical threshold has to be set. This threshold can have a major impact on the analysis results, but is not easy to define.

For these reasons we propose a “top-down” approach, in which we find a summary statistic for each GO term; this summary statistic can be viewed as a new latent variable created from the individual genes in that GO term. These sets of biologically related genes can create a smaller number of new variables with informative descriptions of the biology. If the ontology is accurate, we shall see most of the highly correlated genes being grouped in one set. On one hand, we can reduce the number of variables and alleviate the problem of \( p \gg n \) where \( p \) is the number of genes and \( n \) is the number of samples. On the other hand, we are able to increase interpretability of our findings in ways similar to Fisher’s exact test and GSEA but with increased power. Our procedure, called Domain Enhanced Analysis (DEA), is described as follows:

1. Group genes into each GO term. (Some genes will exist in multiple terms.)

2. Create a new latent variable for each GO term by combining the individ-
ual genes within that GO term. Much can be said about this step.

3. Use the latent variable to test the predictive or discriminating ability of each GO term. Much can be said about this step.

4. Adjust p-values accordingly for multiple testing.

Our major effort was put into finding the best summary statistic for each GO term. Mean, principal component analysis (PCA) and partial least squares (PLS) were a number of options considered. After careful literature review, PLS appeared to be the favorite, because PLS finds a linear combination of the predictor variables $X$ that captures the variation of $X$ related to response $y$, while principal components only explain the variation of $X$ through the variance-covariance structure $X^TX$. The mean is not designed to maximize any particular variation. We conducted a series of simulation studies and analysis on experimental data sets. The results verified that DEA-PLS outperforms DEA-PCA completely when keeping only the first component. DEA-PLS also outperforms DEA-Mean especially when the number of genes grows large for each GO term.

After finding the best summary statistic for each gene set, we proceed to compare DEA-PLS with the classical methods, the Fisher’s exact test and GSEA. The testing procedures in DEA-PLS were done with the two-sample
t-test and the resulting p-values were adjusted by the BH adjustment of Benjamini and Hochberg (1995). We conducted several simulation studies to show that DEA-PLS has greater power than both the Fisher’s exact test and GSEA for detecting significant gene sets. Analysis was also conducted for the two publicly available experimental data sets by Chiaretti et al. (2004) and Golub et al. (1999). Using DEA-PLS, we had some very interesting findings that were missed or not emphasized by the other methods. In general, DEA-PLS increases the interpretability of analysis results on microarray data by focusing on GO term level analysis instead of gene level analysis. Furthermore, it increases the power of detecting important GO terms with comparison to the more standard approach, i.e. the Fisher’s exact test and GSEA. Finally, one key aspect of DEA is that it is adaptive to other analysis techniques. After combining the variables by GO terms using PLS, we basically have a new set of data with fewer but more meaningful variables. We can either build a predictive model with GO terms, or improve our testing procedure with many existing methods for microarray data.

During our development of DEA-PLS, we used the simple two-sample t-test at first to test for important GO terms. Gradually, we found this test inadequate to adjust for GO terms with different number of genes and lack of theoretical justification. We therefore focused on the testing part of DEA-
PLS. Based on assumptions that each non-important gene expression is from a normal distribution and each important gene expression is from a mixture of normal distribution, we derived the distributional information of the first PLS score. Following the same construction of the two-sample t-test, we were able to find the true null distribution. Since the true null distribution has a non-trivial form, several methods of approximation were proposed, including central limit theorem approximation, Edgeworth series approximation and skewed t approximation by matching moments. We compared these methods by simulations and concluded that skewed t approximation performs better than CLT and Edgeworth series approximation.

Understanding the underlying distribution information of our null distribution, we were able to make valid adjustments for our test statistics for GO terms with different sizes. We applied the new test on the simulated data set created previously. The results are slightly better than the original DEA-PLS. In conclusion, by doing the theoretical derivation, we are able to find a test statistic that is more effective than the standard t tests. In the mean time, we find a theoretically justified adjustment for GO terms with larger number of genes $m$ and dramatically reduce the computation time. From our simulation study, we find that the new tests have good power and are effective at finding important GO terms. In our derivation, we focused on cases where the response
variable is binary, specifically \( y = \begin{bmatrix} 0_n^T & 1_n^T \end{bmatrix} \). We were able to extend our test to unbalanced cases for two classes.

Throughout much of our development of DEA-PLS, we used \( y = \begin{bmatrix} 0_n^T & 1_n^T \end{bmatrix} \) to represent the two classes of responses. There has been little literature available discussing the impact of different codings of the response variable in the PLS procedure. In the recent development, majority of time was spent on figuring out the difference of 0/1 or −1/1 coding of the response variable. We focused on how the different coding impact specifically on the new test statistics we created. With some theoretical derivation and simulations, we found that the two codings are in fact equivalent. The findings are being used to extend our test of important GO terms to expression data with less distributional assumptions on the gene expressions.

In summary, below are my original contributions:

1. Understanding the structure of the Gene Ontology and creating the mappings between the Gene Ontology and the microarray data on hand.

2. Finding the best summary statistics for each GO term and comparing a number of options including mean, principal component analysis (PCA) and partial least squares (PLS).

3. Conducting simulation studies to show that DEA-PLS does have greater
power than the Fisher’s exact test and GSEA for detecting significant
gene sets and applying DEA-PLS analysis on two public available exper-
iment data sets by Chiaretti et al. (2004) and Golub et al. (1999).

4. Deriving the distributional information of the first PLS score, based on
assumptions that each non-important gene expression is from a normal
distribution and each important gene expression is from a mixture of
normal distribution.

5. Approximating the true null distribution of our new test statistics us-
ing central limit theorem approximation, Edgeworth series approxima-
tion and skewed t approximation by moments and comparing the three
type of approximation via simulations.

6. Studying the impact of the different coding for y response variable and
using theoretical derivation and simulations to find that the two type of
codings are essentially the same.

7. Extending our test of important GO terms to expression data with less
distributional assumptions on the gene expressions.
Chapter 2

Literature Review

2.1 Introduction of Microarray Technology

Advances in microarray technology have greatly enhanced gene expression studies. In these studies, thousands of genes are monitored and probed on only a few to tens of samples. Gene expression data present both great opportunities and challenges. Patterns of gene expression can be used to determine genes with similar behavior, suggest biomarkers for a specific disease, and propose targets for drug intervention.

There are countless books and papers available in the literature discussing different aspects of microarray gene expression data, from the production,
cleaning and normalization of the data to different analysis methods applied to the data, such as clustering and building classification and prediction models. For starters, McLachlan et al. (2004) and Dudoit et al. (2002) give extensive reviews respectively on microarray technologies as well as different aspects of statistical methods involved in analysis.

Here is a brief summary of the biology related to cDNA and oligonucleotide microarray technology as given in Dudoit et al. (2002, pp. 113-114). “A gene consists of a segment of DNA which codes for a particular protein, the ultimate expression of the genetic information. A deoxyribonucleic acid or DNA molecule is a double-stranded polymer composed of four basic molecular units called nucleotides. Each nucleotide comprises a phosphate group, a deoxyribose sugar, and one of four nitrogen bases. The four different bases found in DNA are adenine (A), cytosine (C), guanine (G), and thymine (T). The two chains of the DNA molecule are held together by hydrogen bonds between nitrogen bases, with base-pairing occurring according to the following rule: G pairs with C, and A pairs with T. While a DNA molecule is built from a four-letter alphabet, proteins are sequences of twenty different types of amino acids. The expression of the genetic information stored in the DNA molecule occurs in two stages: (i) transcription, during which DNA is transcribed into messenger ribonucleic acid or mRNA, a single-stranded complementary copy of the base sequence in
the DNA molecule, with the base uracil (U) replacing thymine; (ii) translation, during which mRNA is translated to produce a protein. The correspondence between DNAs four-letter alphabet and a proteins twenty-letter alphabet is specified by the genetic code, which relates nucleotide triplets to amino acids. Different aspects of gene expression can be studied using microarrays, such as expression at the transcription or translation level, and subcellular localization of gene products. To date, attention has focused primarily on expression at the transcription stage, i.e., on mRNA or transcript levels. Microarrays derive their power and universality from a key property of DNA molecules described above, complementary base-pairing, and the term hybridization is used to refer to the annealing of nucleic acid strands from different sources according to the base-pairing rules.”

Two of the more commonly used microarray systems include the cDNA microarrays developed in the Brown and Botstein labs at Stanford and high-density oligonucleotide chips from the Affymetrix Company. The Chipping Forecast (The-Chipping-Forecast, 1999) includes many reviews about the biology and technology of cDNA microarrays and oligonucleotide chips. For data analysis purposes, we use the term microarrays in a more general sense to refer to both cDNA and oligonucleotide arrays.

Dudoit et al. (2002, p. 114) described the making of the cDNA microar-
rays as following: “cDNA microarrays consist of thousands of individual DNA sequences printed in a high-density array on a glass microscope slide using a robotic arrayer. The relative abundance of these spotted DNA sequences in two DNA or RNA samples may be assessed by monitoring the differential hybridization of the two samples to the sequences on the array. For mRNA samples, the two samples or targets are reverse-transcribed into cDNA, labeled using different fluorescent dyes (usually a red-fluorescent dye, Cyanine 5 or Cy5, and a green-fluorescent dye, Cyanine 3 or Cy3), then mixed in equal proportions and hybridized with the arrayed DNA sequences or probes (following the definition of probe and target adopted in The Chipping Forecast (1999)). After this competitive hybridization, the slides are imaged using a scanner and fluorescence measurements are made separately for each dye at each spot on the array. The ratio of the red and green fluorescence intensities for each spot is indicative of the relative abundance of the corresponding DNA probe in the two nucleic acid target samples.”

The process of acquiring microarray data is very complicated and requires sufficient quality control. It also involves image cleaning and processing to obtain accurate intensity for data analysis. Once we obtain the data, a normalization procedure is required to identify and remove sources of systematic variation other than differential expression in the measured fluorescence inten-
sity. However, this is not the focus of our research. The data we used are already preprocessed. We then standardize the data by subtracting their mean and scaling them to the same variance for each gene. For further reading, Schuchhardt et al. (2000) compared different normalization strategies in the context of cDNA microarrays and Yang et al. (2001) also gave great details on normalization.

The most common analyses applied to microarray data is clustering. There are two distinct clustering problems. We can either focus on clustering of tissue samples on the basis of the genes or clustering of genes on the basis of the tissues. The result of clustering the tissues samples can be used to discover and understand new subclasses of diseases. For example, Ross et al. (2000) classified sixty human cancer cell lines in his paper. Two different types of human acute leukemia were distinguished by Golub et al. (1999). More recently, Bullinger et al. (2004) identified prognostic subclasses in acute myeloid leukemia and Lapointe and Li (2004) found tumor subtypes of prostate cancer that could improve prognostication and treatment stratification of such disease. On the other hand, the clusters of genes obtained as a result can be used to search for genetic pathways or groups of genes that might be regulated together. The insights gained then can be used to create pathways or functional annotation for genes such as the Gene Ontology (Ashburner et al., 2000). McLachlan et al.
(2004) provide a good overview of different clustering methods and Jiang et al. (2004) give a detailed review of all popular clustering methods.

We do not go into the details for clustering methods, because it is not the focus of our research. Our research mainly concentrates on testing and detection of differentially expressed genes, and more importantly the biological insights into the pathways and functions affected by the disease. Next, we will be giving a detailed review on detection of differentially expressed genes.

The basic approach for identifying differentially expressed genes consists of two steps: (1) computing a test statistic for each gene and ordering the features from those showing the most signal of interest to those showing the least; (2) obtaining adjusted p-values with a multiple testing procedure to allow us to draw a significance cut-off somewhere along this ranking. We will review the literature from these two aspects.

2.2 Tests for Differentially Expressed Genes

For starters, Cui and Churchill (2003) did a review on statistical tests for differential expression in cDNA microarrays. They mainly focused on regular t-tests to compare two conditions and analysis of variance (ANOVA) for more than two conditions. Pan (2002) also did a review comparing the regular t-
test, a regression modeling approach (Thomas et al., 2001), and a mixture model approach (Pan et al., 2001) (Pan et al., 2003). It is pointed out that all these three methods are based on using the two-sample t-statistic or its minor variation. Furthermore, the mixture model (Pan et al., 2001) also follows the basic idea of Empirical Bayes (EB) (Efron et al., 2001) and the Significance Analysis of Microarray (SAM) method (Tusher et al., 2001). Before getting into the pros and cons of these methods, first we will give an overview of each method.

2.2.1 Fold Change

Before statisticians got involved with gene expression data, biologists used a method called a “fold” change to identify differentially expressed genes. Fold change evaluates the ratio of the averages of the expression level between two classes and considers all genes that differ by more than an arbitrary cutoff value to be differentially expressed. For example, if the cut-off value chosen is a one-fold difference, genes are considered to be differentially expressed if the expression level under one condition is over one time greater or less than under the other condition. Cui and Churchill (2003) pointed out that fold change is not a statistical test and there is no associated level of confidence in the designation of a gene as being differentially expressed or not differentially
expressed. Also, this method does not incorporate the variance of the replicates in each class. Hence it is now regarded unreliable and inefficient. The fold change can be found in some earlier literature on microarray data; see Schena et al. (1996) and DeRisi et al. (1997).

2.2.2 T-test

For testing for differentially expressed genes with two classes, the t-test is a natural choice of statistical test. Let us suppose that $x_{ij}$ is the expression level of gene $i$ in array $j$ ($i = 1, \cdots, M; j = 1, \cdots, n_0, n_0 + 1, \cdots, n_0 + n_1$). Suppose that the first $n_0$ and last $n_1$ arrays are obtained under the two conditions respectively. A general statistics model is

$$x_{ij} = a_i + b_i y_j + \epsilon_{ij},$$

(2.1)

where $y_j = 0$ for $1 \leq j \leq n_1$ and $y_j = 1$ for $n_0 + 1 \leq j \leq n_0 + n_1$, and $\epsilon_{ij}$ are independently identically distributed random errors with mean 0. Hence determining whether a gene has differential expression is equivalent to testing the null hypothesis

$$H_0: b_i = 0 \text{ vs } H_1: b_i \neq 0.$$ 

There are several versions of the two sample t-test, depending on whether
the gene expression levels have equal variance under the two conditions. Let us assume that we have the expression level with two independent samples with unequal variance. The sample means and variances of $x_{ij}$ for gene $i$ under two conditions can be computed as

$$
\bar{x}_{i,(0)} = \frac{\sum_{j=1}^{n_0} x_{ij}}{n_0}, \quad \bar{x}_{i,(1)} = \frac{\sum_{j=n_0+1}^{n_0+n_1} x_{ij}}{n_1}
$$

and

$$
s^2_{i(0)} = \frac{\sum_{j=1}^{n_0} (x_{ij} - \bar{x}_{i,(0)})^2}{n_0 - 1}, \quad s^2_{i(1)} = \frac{\sum_{j=n_0+1}^{n_0+n_1} (x_{ij} - \bar{x}_{i,(1)})^2}{n_1 - 1}.
$$

Therefore, the t-statistic is

$$Z_i = \frac{\bar{x}_{i,(0)} - \bar{x}_{i,(1)}}{\sqrt{s^2_{i(0)}/n_0 + s^2_{i(1)}/n_1}}, \quad \text{(2.2)}$$

Under $H_0$ and the normality assumption for $x_{ij}$, $Z_i$ approximately has a t-distribution with degrees of freedom

$$d_i = \frac{(s^2_{i(0)}/n_0 + s^2_{i(1)}/n_1)^2}{(s^2_{i(0)}/n_0)^2/(n_0 - 1) + (s^2_{i(1)}/n_1)^2/(n_1 - 1)}.$$

This is the Welch t-test (Welch, 1947).

The Welch t-test is commonly applied to test for differentially expressed gene expressions. For example, Callow et al. (2000) conducted a standard t-
test for each gene with error variance estimated for each gene from replicated experiments. This type of gene-specific t test is not affected by heterogeneity in variance across genes because it only uses information from one gene at a time. However, the variance estimate from each gene might be unstable because of small sample sizes when we only have tens of arrays recorded. To avoid the instability of this variance estimate, Arfin et al. (2000) chose to compute a global t-test using an estimate of error variance that is pooled across all genes under the assumption that the variances are homogeneous between different genes. However, this global t-test approach might suffer serious biases if the error variance is not truly constant for all genes.

2.2.3 Modified t-test

Neither the Welch t-test with gene-specific nor pooled variance mentioned above are perfect. The former is subject to instability of variance estimates when the sample size is small. The latter is more stable but can be biased if the assumption of homogeneity does not hold. A lot of research has been done in this area to create a more reliable modified t-test. For example, the regularized t-test by Baldi and Long (2001), significant analysis of microarrays (SAM) by Tusher et al. (2001) and the regression model by Thomas et al. (2001).
The regularized t-test (Baldi and Long, 2001) combines information from gene-specific variance and global average variance estimates by using a weighted average of the two as the denominator for a gene-specific t-test. Following the notation from Equation 2.2, we define the gene specific standard error as

\[ SE_i = \sqrt{\frac{s_i^2(0)}{n_0} + \frac{s_i^2(1)}{n_1}} \]

and SE be the standard error computed by combining data across all genes

\[ SE = \sqrt{\frac{s^2(0)}{n_0 \cdot M} + \frac{s^2(1)}{n_1 \cdot M}} \]

where

\[ \bar{x}_{(0)} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{n_0} x_{ij}}{n_0 \cdot M}, \bar{x}_{(1)} = \frac{\sum_{i=1}^{M} \sum_{j=n_0+1}^{n_0+n_1} x_{ij}}{n_1 \cdot M} \]

and

\[ s^2(0) = \frac{\sum_{i=1}^{M} \sum_{j=1}^{n_0} (x_{ij} - \bar{x}_{(0)})^2}{n_0 \cdot M - 1}, s^2(1) = \frac{\sum_{i=1}^{M} \sum_{j=n_0+1}^{n_0+n_1} (x_{ij} - \bar{x}_{(1)})^2}{n_1 \cdot M - 1}. \]

The regularized t-test statistic for gene \( i \) is defined as

\[ T(\text{regularized})_i = \frac{\bar{x}_{i,(0)} - \bar{x}_{i,(1)}}{\sqrt{n_0SE^2 + (n-1)SE^2}} \] (2.3)
where $v_0$ is a tunable parameter that determines the relative contributions of gene-specific and global variance and $n = n_0 = n_1$ is the number of replicate measurements for each condition assuming balanced design (unbalanced cases can be easily extended). The degree of freedom of this regularized t-test is $v_0 + n$. Baldi and Long (2001) model the gene expressions by independent normal distributions, parameterized by corresponding means and variance with hierarchical prior distribution. Then, the point estimates for both parameters and hyper-parameters and regularized expressions for the variance of each gene are derived by combining the empirical variance with local background variance associated with neighboring genes.

Using a different approach from the regularized t-test, Tusher et al. (2001) proposed a modified t-test, called significant analysis of microarrays (SAM), by adjusting the gene specific variance with a constant $s_0$. The test statistic for each gene is defined as

$$S_i = \frac{\bar{x}_{i(0)} - \bar{x}_{i(1)}}{SE_i + s_0}$$

(2.4)

To compare values of $S_i$ across all genes, the distribution of $S_i$ should be independent of the level of gene expression. At low expression levels, variance in $S_i$ can be high because of small values of $SE_i$. To ensure that the variance of $S_i$ is independent of gene expression, a small positive constant $s_0$ is added to the
gene specific variance $SE_i$. The coefficient of variation of $S_i$ was computed as a function of $SE_i$ across the data. The value for $s_0$ was chosen to minimize the coefficient of variation. Tusher et al. (2001) suggested estimating the null distribution directly by permuting the data. This approach takes full advantage of the existence of replicated data and tends to be more robust.

Thomas et al. (2001) proposed a regression modeling approach. In the original formulation, as pointed out in Pan (2002), they combined the data preprocessing and testing procedure. Fundamentally, the regression model is the same as our Equation 2.1. Their test statistic is defined as

$$Z_{\text{regression}}(i) = \frac{\hat{b}_i}{\sqrt{\text{var}(\hat{b}_i)}}.$$  \hspace{1cm} (2.5)

They suggested estimating $(a_i, b_i)$ using the weighted least square method, where the weights are the heterogeneity factors that are introduced to account for variations in preparing multiple mRNA samples for each gene, and then estimate the variance of $\hat{b}$ using the robust or sandwich variance estimator. Thomas et al. (2001) observed that the result based on $Z_{\text{regression}}(i)$ in Equation 2.5 is close to that of using the t-statistics $Z_i$ in Equation 2.2. A theoretical explanation was not given in their paper, but can be found in Pan (2002).
Pan (2002) verified that the weighted least-squares estimate of $b_i$ is in fact:

$$\hat{b}_i = \bar{x}_{i(0)} - \bar{x}_{i(1)}.$$ 

And the robust variance estimator of $\hat{b}_i$ is

$$\text{Var}(\hat{b}_i) = \frac{s_i^2 n_0}{n_0} - \frac{1}{n_0} + \frac{s_i^2 n_1}{n_1} - \frac{1}{n_1}.$$ 

$s_i, n_0, n_1$ are defined in Equation 2.2. It is easy to see that the statistic $Z_{(\text{regression})i}$ has a similar form to the Welch t statistic $Z_i$ with a different estimation of the variance. Instead of using the unbiased sample variance, the maximum likelihood estimator of the variance (under the normality assumption for $x_{ij}$) is used for $Z_{(\text{regression})i}$. Pan (2002) pointed out that $Z_{(\text{regression})i}$ and $Z_i$ are asymptotically equivalent as both $n_0$ and $n_1$ go to the infinity. However, for small $n_0$ and $n_1$, $Z_i$ is preferred due to the unbiasedness of the variance estimator. Furthermore, the normality assumption for $Z_{(\text{regression})i}$ is based on assuming large $n_0$ and $n_1$, which does not hold in many microarray experiments. Therefore, $Z_{(\text{regression})i}$ might not work well in practice.

The three methods mentioned above are based on the standard t statistics with different variations. SAM is more robust with a liberal assumption by estimating its null distribution with permutation. On the other hand, the
regression model approach uses too strong an assumption which might not be working in practice. The regularized t statistics use a Bayesian framework to combine the gene-specific variance with the global gene variance, which is a step forward from the standard t statistics.

Other adjustments have also been proposed, such as the t-test proposed by Cui et al. (2005), constructing improved estimators of variance from an ensemble of individual variance estimators by shrinking them towards common corrected geometric means. It is another way of adjusting the gene specific variance based on the global gene variance. According to the simulation in the paper, their new test shows best or nearly best power for detecting differentially expressed genes among methods, including regularized t and SAM.

2.2.4 Mixture Model Approach

SAM by Tusher et al. (2001) suggested estimating the null distribution from permutation, which overcomes the common problem of strong assumption on the null distribution with modified t-tests and regression methods. Other approaches also have been proposed to estimate the null distribution. Tests under the mixture model framework are the more popular ones.

The empirical Bayes (EB) by Efron et al. (2001) is among the earliest de-
velopment which built the foundation of the mixture model approach. In their paper, the test statistic is defined similarly to SAM as Equation 2.4,

\[ S_i = \frac{\bar{x}_{i(0)} - \bar{x}_{i(1)}}{\text{SE}_i + s_0} \]  

(2.6)

except that \( s_0 \) is fixed to be the 90th percentile of all values of \( \text{SE}_i \) in the data. Then a simple inference model is introduced that can be applied to most comparative experiments: a gene is either affected or unaffected by the treatment of interest, giving two possible distributions for test statistics \( S \). Let

\[ p_1 = \text{probability that a gene is affected} \]

\[ p_0 = \text{probability that a gene is unaffected} \]

and

\[ f_1(s) = \text{the density of } S \text{ for affected genes} \]

\[ f_0(s) = \text{the density of } S \text{ for unaffected genes.} \]

Then

\[ f(s) = p_0f_0(s) + p_1f_1(s) \]

is the mixture density of the two populations.

The distribution of \( f(s) \) is suggested being estimated directly from all ob-
served test statistics $S$ (Efron et al., 2001). Without a parametric assumption
of the null distribution, $f_0(s)$ is also estimated from data with a little trick.
The null statistics $S_i$ is not calculated by average difference of expression level
of samples between two classes, but the average difference of expression level
for samples within each class. For example, suppose we have two classes of $n_0$
and $n_1$ samples, class I and class II. The samples under each class are evenly
split into two groups at random, say “A” and “B” for class I, “C” and “D”
for class II. The null scores are defined as the difference between groups $A \cup C$
versus $B \cup D$ of each gene expression. The basic idea is to recover the null
distribution from differences that negate treatment effects.

After $f_0(s)$ and $f(s)$ are obtained, an application of Bayes’ rule to the
mixture model gives the posteriori probabilities $p_1(S)$ and $p_0(S)$ that a gene
with score $S$ was affected or unaffected by the treatment:

$$p_1(S) = 1 - p_0 f_0(S) / f(S)$$

$$p_0(S) = p_0 f_0(S) / f(S).$$  \hspace{1cm} (2.7)

The ratio of $f_0(S)/f(S)$ can be estimated directly from the empirical distribu-
tions of $f_0(S)$ and $f(S)$. The constraint that $p_1(S)$ is nonnegative for all $S$
restricts $p_0$ and $p_1$ as

$$p_1 \geq 1 - \min_s \{f_0(S)/f(S)\}.$$ 

and

$$p_0 \leq 1 - \min_s \{f_0(S)/f(S)\}.$$ 

The value of $p_0$ may be taken as its estimated maximum value or more conservatively with $p_0 = 1$ based on the assumption that most of the genes are not differentially expressed in the data.

There are different variations of estimating the distribution of true expression score “S”. In EB, no parametric assumptions are given. In other approaches with more parametric restrictions, Lee et al. (2000) used a normal theory version and Newton et al. (2001) focused on Gamma models. Lonnstedt and Speed (2002) proposed a lognormal-normal model-based empirical Bayes method giving posterior probabilities of differential expression. Allison et al. (2002) used a mixture model approach, but they worked with $p$ values instead of $Z$. In their paper, it is suggested using uniform distribution as the null distribution $f_0(p)$ and a mixture of beta distributions to approximate the distribution of observed $p$-values $f(p)$.

Among other approaches, the mixture model methods (MMM) proposed
by Pan et al. (2001) is quite influential. It follows the idea of SAM and EB, besides taking full advantage of the existence of replicated data by estimating the null distribution directly. A key step is to construct the null statistics as following:

\[ z_i = \frac{x_i^{(0)} p_i - x_i^{(1)} q_i}{\sqrt{\frac{s_i^{2(0)}}{n_0} + \frac{s_i^{2(1)}}{n_1}}} \] (2.8)

where \( x_i^{(0)} = (x_{i1}, \cdots, x_{in_0}) \), \( x_i^{(1)} = (x_{i,n_0+1}, \cdots, x_{i,n_0+n_1}) \), \( p_i \) is a random permutation of a column vector containing \( n_0/2 \) 1s and -1s respectively, and \( q_i \) is a random permutation of a column vector containing \( n_1/2 \) 1s and -1s. And \( Z_i \) is defined in Equation 2.2. Suppose that the probability density functions of \( z_i \) and \( Z_i \) are respectively \( f_0 \) and \( f \). Then under \( H_0 \), \( z_i \) and \( Z_i \) have the same distribution \( f_0 = f \). Using \( z_i \)'s and \( Z_i \)'s, the distributions of \( f_0 \) and \( f \) can be estimated respectively. For any given \( Z \), the likelihood ratio test statistic is created as:

\[ LR(Z) = \frac{f_0(Z)}{f(Z)} \] (2.9)

to test for \( H_0 \). A small value of LR\( (Z) \), say LR\( (Z) < c \), provides evidence to reject \( H_0 \). The cut-off point \( c \) is determined such that the type I error rate is

\[ \frac{\alpha}{m} = \int_{LR(Z) < c} f_0(z) dz, \]

where \( \alpha \) is the genome-wide significance level and \( m \) is the number of genes in
the data.

As for how to estimate $f_0$ and $f$, Pan et al. (2001) proposed to use a Normal mixture to model each distribution:

$$f_0(z; \Omega_{g_0}) = \sum_{i=1}^{g_0} \pi_i \phi(z; \mu_i, V_i),$$

where $\phi(.; \mu_i, V_i)$ denotes the normal density function with mean $\mu_i$ and variance $V_i$ and $\pi_i$s are mixing proportions. $\Omega_{g_0}$ represents all unknown parameters $\{ (\pi_i, \mu_i, V_i : i = 1, \cdots, g_0 \}$ in a $g_0$-component mixture model. The number of components can be selected adaptively. Similarly, a Normal mixture model can be used for $f$. In Pan et al. (2001), the normal mixture is fitted by maximum likelihood using the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). To determine the number of components $g_0$, they use the Bayesian Information Criterion (BIC) (Schwartz, 1978).

It is essentially the same how EB and MMM define their null statistics $z$. The main difference lies in the different ways of constructing the test statistics as well as how $f_0$ and $f$ are estimated. EB simply uses the empirical distribution estimated from data, while MMM estimates both distributions using a mixture of Normals.

Further improvement on this approach is proposed by Zhao and Pan (2003).
They pointed out that MMM relies on the implicit assumption that the numerator and denominator of $Z_i$ and $z_i$ are independent, which in general does not hold especially for $z_i$. To solve this problem, they proposed a new null statistic $z_i^*$. Specifically,

$$z_i^* = \frac{W_i(0) - W_i(1)}{\sqrt{\frac{s_{i(0)}^2}{k_0} + \frac{s_{i(1)}^2}{k_1}}} \quad (2.10)$$

where $k_0 = n_0/2$, $k_1 = n_1/2$, $W_{ik} = (x_{i,k} - x_{i,k_0+k})/2$ for $k = 1, \ldots, k_0$ and $W_{ik} = (x_{i,k_0+k} - x_{i,k_0+k_1+k})/2$ for $k = k_0 + 1, \ldots, k_0 + k_1$ and

$$s_{i(0)}^2 = \sum_{k=1}^{k_0} W_{ik} - \bar{W}_{i(0)}$$

$$s_{i(1)}^2 = \sum_{k=k_0+1}^{k_0+k_1} W_{ik} - \bar{W}_{i(1)}$$

Under this setup, the new test statistics $Z_i^*$ have the same denominator as that of $z_i^*$ and has the same numerator as the original $Z_i$ in Equation 2.2. Specifically,

$$Z_i^* = \frac{\bar{x}_{i(0)} - \bar{x}_{i(1)}}{\sqrt{\frac{s_{i(0)}^2}{k_0} + \frac{s_{i(1)}^2}{k_1}}} \quad (2.11)$$

By doing this, under the normality assumption on $x_{ij}$, Zhao and Pan (2003) proved that the numerator and the denominator of $Z_i^*$ and $z_i^*$ are independent.
respectively. Furthermore, under $H_0$, $Z_i^*$ and $z_i^*$ have the same distribution. The disadvantage is that the new test has a loss of power since the sample sizes in each group have been reduced to a half of the original. $(n_0 + n_1)/2$ pair-wise differences are used to estimate the two sample variances in the denominator of $Z_i^*$ and $z_i^*$, leading to reduced degrees of freedom $(n_0 + n_1)/2 - 2$, in contrast to the original $n_0 + n_1 - 2$. According to the simulations in that paper, the modified test statistics worked better than original MMM. In Zhao and Pan (2003), modification was also proposed to increase degrees of freedom when the data has an unbalanced design and the replicates under one condition are much larger than those under the other.

### 2.2.5 Optimal Discovery Procedure

In the most recent literature, Storey et al. (2007) proposed the Optimal Discovery Procedure (ODP) for large scale significance testing. The ODP is defined as the testing procedure that maximizes the expected number of true positives (ETP) for each fixed level of expected number of false positives (EFP). The FDR procedure used in ODP is proposed by Storey (2002), which will be covered later. Here, we are focusing on the ODP statistics proposed.

The ODP is based on one of the fundamental ideas behind individual significance test: the Neyman-Pearson Lemma. Given a single set of observed data,
the optimal single testing procedure is based on the statistic:

\[ S_{NP}(data) = \frac{\text{probability of the data under the alternative distribution}}{\text{probability of the data under the null distribution}}. \]

The null hypothesis is then rejected if the statistic \( S_{NP} \) exceeds some cut-off chosen to satisfy an acceptable Type I error rate. This Neyman-Pearson procedure is optimal because it is “most powerful”, meaning for each fixed Type I error rate, there does not exist another rule that exceeds this one in power.

The ODP statistic is written similarly to the NP statistic, except that the ODP considers the data for a single feature evaluated at all true probability density functions instead of considering the data evaluated at its own alternative and null probability density functions. Let “data\(_i\)” be the data for the \( i \)th (gene) being tested. The ODP statistic for gene \( i \) is calculated as:

\[ S_{ODP}(data_i) = \frac{\text{sum of probability of data}_i \text{ under each true alternative distribution}}{\text{sum of probability of data}_i \text{ under each true null distribution}}. \]

(2.12)

For a fixed cut-off chosen to attain an acceptable EFP (or FDR level), each null hypothesis is rejected if its ODP statistic exceeds the cut-off.

The mathematical formulations can be defined following the setup in Equation 2.1, where \( M \) tests are performed on observed data sets \( x_1, \cdots, x_M \), where each significance test consists of \( n = n_0 + n_1 \) observations so that each
\( x_i = (x_{i1}, \cdots, x_{in}) \). Assume that significance test \( i \) has null probability density function \( f_i \) and alternative density \( g_i \) and suppose that the null hypothesis is true for tests \( i = 1, 2, \cdots, m_0 \) and the alternative is true for \( i = m_0 + 1, \cdots, M \).

The ODP statistic for each gene \( i \) can be define mathematically as:

\[
S_{ODP}(x_i) = \frac{f_1(x) + f_2(x) + \cdots + f_{m_0}(x) + g_{m_0+1}(x) + g_{m_0+2}(x) + \cdots + g_M(x)}{f_1(x) + f_2(x) + \cdots + f_{m_0}(x)}.
\]

(2.13)

Null hypothesis \( i \) is rejected if and only if \( S_{ODP}(x_i) \geq \lambda \), where \( \lambda \) is chosen to satisfy an acceptable EFP or FDR level. Note that in fact equation (2.13) = 1+equation (2.12).

For the case of detecting differentially expressed genes, Storey et al. (2007) estimate the ODP with the assumption that \( x_{ij} \) comes from a Normal distribution with mean \( \mu_{i0} \) for \( y_j = 0 \) or \( \mu_{i1} \) for \( y_j = 1 \) and variance \( \sigma_i^2 \). Under this assumption, the likelihood of data \( x \) with mean \( \mu \) and variance \( \sigma^2 \) is written as

\[
\phi(x; \mu, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp \left( -\frac{\sum_{j=1}^{n}(x_j - \mu)^2}{2\sigma^2} \right),
\]

where \( \phi \) is the normal probability density function. Therefore for our general ODP estimates, we define

\[
f_i(x) = \phi(x_i; \mu_i, \sigma_i^2), g_i(x) = \phi(x_{0i}; \mu_{0i}, \sigma_i^2)\phi(x_{1i}; \mu_{1i}, \sigma_i^2).
\]
These densities can be estimated in a straightforward way using \((\hat{\mu}_i, \hat{\sigma}_i^2)\) the maximum likelihood estimates under the constraints of the null hypothesis, and 
\((\hat{\mu}_0, \mu_1, \hat{\sigma}_i^2)\) the unconstrained maximum likelihood estimates.

The overall algorithm proposed includes computing the observed statistics 
\(S_{\text{ODP}}(x_i)\) in Equation 2.13 from the data and the null statistics from re-
sampled data using bootstrap. Finally, using these observed and null statistics, 
the cut off is set according to the estimated positive false discovery rate (q-
value) (Storey, 2002). The estimation of q-values will be reviewed later. Storey
\textit{et al.} (2007) compared ODP to methods including SAM (Tusher \textit{et al.}, 2001), 
standard t-test, modified t-test with adjusted variance components (Cui \textit{et al.}, 
2005) and model-based empirical Bayes methods (Lonnstedt and Speed, 2002). 
According to the results of simulation and data analysis, ODP shows a better 
detection rate for a fixed false discovery rate.

2.2.6 Summaries

We reviewed the literature regarding tests for detecting differentially ex-
pressed genes. The focus is on the most influential methods, such as different 
variations of t-test, mixture model methods, empirical Bayes, SAM and the 
most recent ODP. There are also other methods proposed in various paper, 
which we will not cover in detail. For example, the standard nonparametric
Wilcoxon rank-sum test. This type of rank test usually has a loss of power. Efron and Tibshirani (2002) pointed out that the standard Wilcoxon rank-sum tests did not perform well at all for data without replication and expected to find 161 false detections out of 3226 genes even if none of the genes are actually significant. Research has been done by Lee et al. (2005b) to improve the standard Wilcoxon rank-sum test. They proposed generalized rank-sum test for two group comparisons and generalized signed rank tests and sign tests. In the most recent paper, Gottardo et al. (2006) developed a robust Bayesian hierarchical model for detecting differential expression. Their method focused on dealing with replicated cDNA microarray data containing outliers.

With so many methods out there, the key is to understand how to apply the right method to the right data. In general, for microarray data, sample sizes can be quite small. Under this situation, tests with strong parametric assumptions are not preferable, i.e., regular t-test, regression model (Thomas et al., 2001), regularized t-test (Baldi and Long, 2001). Although these tests are valid asymptotically, their assumptions are usually violated with real experimental data. For this reason, the nonparametric approaches are much more appealing in practice, such as empirical Bayes (Efron et al., 2001), SAM (Tusher et al., 2001), MMM (Pan et al., 2001) (Zhao and Pan, 2003). These methods are considered nonparametric, in the sense that the null distribution does not
come from a strict parametric family like t distribution, but is estimated by permutation of the data (SAM), by empirical distribution from the data (EB) or by approximation from a powerful family of distributions like mixture of normals (MMM). In terms of these three popular nonparametric approaches, Pan (2002) pointed out that interpretation of the result in terms of significance level for EB may be more conservative in general, because the lower bound of the posterior probability is estimated and given. As for SAM, it is best suited to detect a small number of genes with differential expressions. On the other hand, the improved version of MMM (Zhao and Pan, 2003) increases the power of detecting differentially expressed genes compared to the original MMM (Pan et al., 2001)

The ODP proposed by Storey et al. (2007) approaches the problem from another perspective. Their procedure focuses on maximizing (optimizing) the expected number of true positives (ETP) for each fixed level of expected number of false positives (EFP). Sure enough, comparing with other leading methods, ODP detects the most differentially expressed gene with fixed level of EFP or false discovery rate for normally distributed data. Storey et al. (2007) elaborated this finding by both simulation and experimental data sets.
2.3 Adjustment for Multiple hypotheses Testing

The multiple testing adjustment is very important for correctly detecting differentially expressed genes, because most microarray data consist of thousands of genes, which means that thousands of tests have to be done simultaneously. In the meantime, we want to be able to detect more true differentially expressed genes while sacrificing some false discoveries. The usual approach of multiple testing adjustment by controlling family wise error rate may be too conservative under this situation. Hence the introduction of the concept of controlling false discovery rate.

The general setup for simultaneous hypothesis testing begins with a collection of null hypotheses

\[ H_1, H_2, \ldots, H_N, \]

with their respective alternative hypotheses

\[ H_{i1}, H_{i2}, \ldots, H_{iN}, \]
and corresponding test statistics, possibly not independent,

\[ Y_1, Y_2, \ldots, Y_N, \]

and their p-values, \( P_1, P_2, \ldots, P_N \), with \( P_i \) measuring how strongly \( y_i \), the observed value of \( Y_i \), contradicts \( H_i \), for instance \( P_i = \text{prob}_{H_i}(|Y_i| > |y_i|) \). Some like to work with z-values instead of the \( Y_i \)'s or \( P_i \)'s by a simple transformation function,

\[
z_i = \Phi^{-1}(P_i), i = 1, 2, \ldots, N;
\]

where \( \Phi \) is the standard normal cumulative distribution function. If \( H_i \) is exactly true then \( z_i \) will have a standard normal distribution

\[
z_i|H_i \sim N(0, 1).
\]

Next we will review the literature of different type of adjustment for multiple testing in testing for differential gene expressions.

Table 2.1: Contingency table to show relations between observed and real hypotheses.

<table>
<thead>
<tr>
<th></th>
<th># not rejected</th>
<th># rejected</th>
<th>totals</th>
</tr>
</thead>
<tbody>
<tr>
<td># true H</td>
<td>( N_{00} )</td>
<td>( N_{10} )</td>
<td>( N_0 )</td>
</tr>
<tr>
<td># non-true H</td>
<td>( N_{01} )</td>
<td>( N_{11} )</td>
<td>( N_1 )</td>
</tr>
<tr>
<td>totals</td>
<td>( N - R )</td>
<td>( R )</td>
<td>( N )</td>
</tr>
</tbody>
</table>
2.3.1 Controlling the Family-Wise Error Rate

The early researchers managed to control the family wise error rate (FWER) while adjusting for multiple hypothesis. With a total of $N$ hypotheses, let $N_0$ and $N_1$ be number of the true and false null hypotheses, and $N_{11}$, $N_{10}$, $N_{01}$, $N_{00}$ be the number of true positives, false positives, false negatives and true negatives see Table 2.1. $R = N_{10} + N_{11}$ is denoted to be the total number of rejections observed. Then the FWER is the probability of yielding one or more false positives out of all $N$ hypotheses tested written as (McLachlan et al., 2004):

$$FWER = \text{pr}\{N_{10} \geq 1\}. \quad (2.15)$$

A popular method for controlling the FWER is the Bonferroni method. The test of each $H_i$ is controlled so that the probability of a Type I error is less than or equal to $\alpha/N$ for some $\alpha$. This ensures that the overall FWER is less than or equal to $\alpha$. The Bonferroni method is generally regarded as too conservative. Holm (1979) proposed a less conservative step-up procedure that orders the p-values and makes successively smaller adjustments. Let the ordered p-values be denoted as $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(N)}$. Then the adjusted p-values are calculated by:

$$p_{(1)}^{\text{adj}} = N \times p_{(1)},$$
\[ p_{(j)}^{\text{adj}} = \max\{p_{(j-1)}, (N - j + 1) \times p_{(j)}\}, (1 < j \leq N). \]

Other methods of controlling FWER include a step-down algorithm, a resampling-based procedure proposed by Westfall and Young (1993), which is used to adjust for multiplicity by controlling the FWER without assuming null distribution of each test statistic, and similarly the permutation-based adjustment by Dudoit et al. (2002).

### 2.3.2 False Discovery Rate

The issue with controlling FWER is that they control the probability of at least one false positive regardless of the number of hypotheses being tested. It is too conservative and will miss a lot of true findings when the number of hypotheses \( N \) becomes large. Especially for microarray studies, scientists generally do not want to miss any interesting findings while they can cope with a small proportion of false positives.

Benjamini and Hochberg (1995) proposed a new multiple hypothesis testing error measure called the false discovery rate (FDR). With our previous notation, \( N_0 \) and \( N_1 \) are the numbers of the true and false null hypotheses, \( N_{11}, N_{10}, N_{01}, N_{00} \) are the numbers of true positives, false positives, false negatives and true negatives and \( R = N_{10} + N_{11} \) is the total number of rejections ob-
served. The proportion of errors committed by falsely rejecting null hypotheses can be viewed through the random variable $Q = N_{10}/R$, the proportion of the rejected null hypotheses which was erroneously rejected. Additionally, $Q = 0$ when $R = 0$, as no error of false rejection can be committed. The FDR is defined as the expectation of $Q$ (Benjamini and Hochberg, 1995):

$$FDR = E(Q) = E \left( \frac{N_{10}}{R} \right)$$

(2.16)

also as

$$FDR = E(Q) = E \left( \frac{N_{10}}{R} \mid R > 0 \right) \text{pr}\{R > 0\}.$$  

(2.17)

Benjamini and Hochberg (1995) pointed out two important properties of this error rate. First, if all null hypotheses are true, the FDR is equivalent to the FWER. Secondly, the FDR is smaller than or equal to the FWER in general and hence it is expected to have a gain in power especially when most of the hypotheses are false.

Benjamini and Hochberg (1995) proposed a procedure proved by induction that controls the FDR at level $\alpha$ when p-values following the null distribution are independent and uniformly distributed.

- Step 1: Let $p_{(1)} \leq p_{(2)} \leq \ldots \leq p_{(N)}$ be the ordered observed p-values.
• Step 2: Calculate

\[ \hat{k} = \arg \max_{1 \leq k \leq N} \{k : p_k \leq \alpha k/N \} . \]

• Step 3: If \( \hat{k} \) exists, reject null hypotheses corresponding to \( p_1, p_2, \ldots, p_k \).

Otherwise, reject nothing.

This procedure, usually called the BH procedure, weakly control the FWER when all p-values are independent, but provides strong control of the FDR.

For further developments, Benjamini and Hochberg (2000) suggested an adaptive procedure that combines the estimation of \( N_0 \) with the BH procedure. Benjamini and Yekutieli (2001) extended the FDR procedure to cases where test statistics are positively correlated.

2.3.3 Positive False Discovery Rate

Benjamini and Hochberg (1995) included the condition \( R > 0 \) for the expectation in Equation 2.17 to allow the FDR to be bounded in the case when all the null hypotheses are true, \( R = 0 \). Storey (2002) argued that, if all the null hypotheses are true, one would want the a measure that take the value to be one, and it is a case of no interest where no test is significant. Under
these considerations, Storey (2002) proposed the positive false discovery rate (pFDR), defined by

\[ pFDR = E(Q) = E \left( \frac{N_{10}}{R} | R > 0 \right). \]  \hspace{1cm} (2.18)

The term ‘positive’ was added to reflect the fact that the equation was conditioned on the event that positive findings have occurred.

To estimate pFDR, Storey (2002) assumed that there are \( N \) hypotheses \( H_i, i = 1, \ldots, N \). They are distributed as independent Bernoulli random variables with \( \Pr(H_i \text{ is null}) = p_0 \) and \( \Pr(H_i \text{ is alternative}) = p_1 \). In his paper, \( \pi_0 \) and \( \pi_1 \) are used to denote these two priori probability. We use \( p_0 \) and \( p_1 \) to keep the notation consistent with the rest of the paper. Assume that \( N \) hypotheses have the same rejection region denoted as \( \Gamma \). Consequently, the pFDR is defined as:

\[ pFDR(\Gamma) = \frac{p_0 \Pr(T \in \Gamma | H = 0)}{\Pr(T \in \Gamma)} \]  \hspace{1cm} (2.19)

where \( \Pr(T \in \Gamma) = p_0 \Pr(T \in \Gamma | H = 0) + p_1 \Pr(T \in \Gamma | H = 1) \). Let the rejection region based on p-values be the form \( \Gamma = [0, \gamma] \) for some \( 0 \leq \gamma \leq 1 \), the pFDR
can be rewritten as:

$$p_{FDR}(\gamma) = \frac{p_0 \Pr(P \leq \gamma | H = 0)}{\Pr(P \leq \gamma)} = \frac{p_0 \gamma}{\Pr(P \leq \gamma)},$$  \hspace{1cm} (2.20)

where $P$ is the random p-value resulting from any test, and is assumed to follow a uniform distribution when $H = 0$

To estimate the value of pFDR, Storey (2002) proposed a conservative estimate of $p_0$,

$$\hat{p}_0(\lambda) = \frac{\# \{p_i > \lambda\}}{(1 - \lambda)N} = \frac{W(\lambda)}{(1 - \lambda)N},$$

for some well chosen $\lambda$, where $p_1, \cdots, p_N$ are the observed p-values and $W(\lambda) = \# \{p_i > \lambda\}$. An estimate of $\Pr(P \leq \gamma)$ is

$$\hat{\Pr}(p \leq \gamma) = \frac{\# \{p_i \leq \gamma\}}{N} = \frac{R(\gamma)}{N},$$

where $R(\gamma) = \# \{p_i \leq \gamma\}$. Therefore, an estimate of $p_{FDR}(\gamma)$ for fixed $\lambda$ is

$$\hat{p}_{FDR}(\gamma) = \frac{\hat{p}_0(\lambda) \gamma}{\Pr(P \leq \gamma)} = \frac{W(\lambda) \gamma}{(1 - \lambda)R(\gamma)},$$  \hspace{1cm} (2.21)

As for the optimal value of $\lambda$, Storey (2002) suggested using bootstrap methods to choose a $\lambda$ that minimize the mean-squared error of the estimate of pFDR.

The key difference between pFDR and FDR by Benjamini and Hochberg
(1995) is that using the BH methods to control false discovery rate at level $\alpha/p_0$ is equivalent to using pFDR to control the false discovery rate at level $\alpha$. Positive FDR gains power over BH method by controlling a smaller error rate ($\alpha \leq \alpha/p_0$).

### 2.3.4 Local False Discovery Rate

Benjamini and Hochberg (1995) and Storey (2002) focused on tail area false discovery rates, and they are both conservative compared to the true false discovery rate. We can transform the $p$-values to $z$-values using Equation 2.14. Suppose that the $N$ $z$-values can be separated into two classes, “Non-significant” or “Significant”, depending on whether $z_i$ is generated according to the null hypothesis, with prior probabilities $p_0$ and $p_1 = 1 - p_0$ for each class and $z_i$ has density either $f_0(z)$ or $f_1(z)$ depending on its class.

\[
p_0 = \text{Prob}\{\text{Non-significant}\}, f_0(z) \text{ density} \mid z \text{ is null}
\]

\[
p_1 = \text{Prob}\{\text{Significant}\}, f_1(z) \text{ density} \mid z \text{ is non-null.}
\]

Hence $z$ follows a mixture density $f(z)$,

\[
f(z) = p_0 f_0(z) + p_1 f_1(z).
\]
According to Bayes theorem the posteriori probability of being in nonsignificant class given \( z \) is defined as the local false discovery rate (Efron and Tibshirani, 2002) (Efron, 2004) (Efron, 2005):

\[
\text{lfdr}(z) = \text{Prob}\{\text{null}|z\} = p_0 f_0(z)/f(z) = f_0^+(z)/f(z),
\]

assuming \( p_0 \) is usually close to 1.

We can rewrite Benjamini and Hochberg (1995)'s FDR using \( z \) value. Let \( F_0(z) \) and \( F_1(z) \) be the cdf's corresponding to \( f_0(z) \) and \( f_1(z) \), define \( F(z) = p_0 F_0(z) + p_1 F_1(z) \). Then the posterior probability of a case being null given that its \( z \)-value “\( Z \)” is less than some value \( z \) is

\[
\text{FDR}(z) = \text{Prob}\{\text{null}|Z \leq z\} = p_0 F_0(z)/F(z),
\]

where FDR here follows the definition FDR by Benjamini and Hochberg (1995).

FDR and lfdr are analytically related by

\[
\text{FDR}(z) = \frac{\int_{-\infty}^{z} \text{lfdr}(Z) f(Z) dZ}{\int_{-\infty}^{z} f(Z) dZ} = E_f\{\text{lfdr}(Z)|Z \leq z\},
\]

where “\( E_f \)” indicating expectations with respect to \( f(z) \). FDR(\( z \)) is the average of \( \text{lfdr}(Z) \) for \( Z \leq z \); FDR(\( z \)) will be less than \( \text{lfdr}(z) \) in the usual situation.
since \( \text{lfdr}(z) \) decreases as \(|z|\) gets large. In other word, FDR underestimates the true false discovery rate, while local fdr find its true estimates for each \( z_i \).

Efron (2005) proposed the usual procedure for identifying differentially expressed genes from a microarray data set:

1. Estimate \( f(z) \) from the observed ensemble of \( z \)-values, normally using nonparametric spline fit to the histogram counts;
2. Assign a null density \( f_0(z) \);
3. Calculate \( \text{lfdr}(z) = f_0^+(z)/f(z) \);
4. Report those cases with \( \text{lfdr}(z_i) \) less than some threshold value as significant, i.e. \( \text{lfdr}(z_i) \leq 0.10 \).

Efron suggested using a standard Poisson general linear model to estimate the density of \( f(z) \). As for estimating \( f_0^+(z) \), he suggested fitting \( \hat{f}_0^+(z) \) as a scaled normal density over the central one-third range of the \( z \)-values. The reason is that he assumes the central peak of the histogram of \( z \)-values must consist mainly of null cases. Hence, using peak of the \( z \)-values should give us a reasonable estimation of the null distribution. Efron’s local false discovery rate is available as R package, “locfdr”.

Other than Efron’s version of local false discovery rate, there are several

Local fdr estimates the true false discovery rate for each test statistic $z_i$ instead of the tail area of false discovery rate $z_i \leq z$. It should be an improvement overall from the regular FDR or pFDR adjustment.

2.4 Gene Annotation Enhanced Analysis

One way to improve the simple univariate analysis is to identify alterations in gene expression at the level of biological pathways or co-regulated gene sets rather than individual genes. One of the most popular pathway knowledge database for genes is called the Gene Ontology (GO; see Ashburner et al. (2000)). The genes can be mapped to three hierarchies of GO terms, Molecular Functions, Cellular Components and Biological Processes. Many methods have been proposed to detect important GO terms based on the significance level of related genes, including the earlier approaches such as Fisher’s exact test (Draghici et al., 2003a) and Gene Set Enrichment Analysis (GSEA) (Subramanian et al., 2005). Among the newer developments in recent literature,
Kim and Volsky (2005) tried to improve the standard GSEA method with the proposition of parametric analysis of gene set enrichment (PAGE). PAGE is found to be more sensitive for finding differentially expressed gene sets than regular GSEA by using a parametric model. Goeman et al. (2004) created a test for one or a group of genes based on empirical Bayesian generalized linear model. Pan (2006) proposed using mixture model to incorporating gene functional annotations in detecting differential gene expressions. Delongchamp et al. (2006) introduced another approach using meta-analysis methods for combining p-values of individual genes. The methods were modified to adjust for correlation among the genes related to each GO term. We are going to review some of these methods in this section.

2.4.1 Fisher’s Exact Test

Draghici et al. (2003a) showed that the number of differentially expressed genes in a given GO term can be modeled by a hypergeometric distribution, and hence a test may be carried out with the distributional information. This hypergeometric test is equivalent to Fisher’s exact test. To apply Fisher’s exact test, gene level testing needs to be carried out first to detect differentially expressed genes using any of the methods discussed in Section 2.2 and Section 2.3. After the differentially expressed genes are identified, a contingency table is cre-
ated as shown in Table 2.2. A standard Fisher’s exact test (Agresti, 1992) can be carried out to test whether any of gene set is being over-represented with differentially expressed genes.

Table 2.2: Contingency table classification of genes according to whether they are identified as differentially expressed (flagged) and whether they are part of the GO term.

<table>
<thead>
<tr>
<th>In GO term</th>
<th>Flagged Genes</th>
<th>Non-flagged Genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>n_{11}</td>
<td>n_{12}</td>
<td></td>
</tr>
<tr>
<td>n_{21}</td>
<td>n_{22}</td>
<td></td>
</tr>
</tbody>
</table>

The Fisher’s exact test is among the earlier approaches trying to summarize the results of gene level analysis to GO terms level. Its obvious advantage is that it is relatively easy to implement once the gene level analysis is done. However, the Fisher’s exact test loses power by focusing only on the number of differentially expressed genes related to each GO term.

2.4.2 Gene Set Enrichment Analysis

Besides Fisher’s exact test, Gene Set Enrichment Analysis (GSEA) (Subramanian et al., 2005) is another commonly used method. The basic idea is to combine information from the genes in each GO term to increase signal relative to noise and improve statistical power, and compare them to a null distribution in which genes are randomly distributed. The null hypothesis of GSEA is that
the rank ordering of the gene set in a given comparison is random with regard to the diagnostic categorization of the samples. The alternative hypothesis is that the rank ordering of the members of the gene set is associated with the specific diagnostic criteria used to categorize the groups of affected individuals. GSEA is a step forward from Fisher’s exact test, because the test is based on the rank ordering of the genes related to each GO term instead of simply the number of differentially expressed genes.

To implement GSEA, initial gene level analysis needs to be carried out on all gene expressions. The genes are ordered by the \( p \)-values obtained from univariate analysis. For each gene set, an enrichment measure, which is a normalized Kolmogorov-Smirnov statistic is computed. Consider the genes \( G_1, \ldots, G_n \), that are ordered by their \( p \)-values from univariate analysis and a GO term \( B \) contain \( m \) members (genes). The score function for each GO term \( B \) is defined (see Figure 2.1):

\[
\text{Score} = \max_{1 \leq j \leq n} \sum_{i=1}^{j} X_i,
\]

where

\[
X_i = -\sqrt{\frac{m}{n-m}}
\]
if $G_i$ is not a member of $B$, or

$$X_i = \sqrt{\frac{n - m}{m}}$$

if $G_i$ is a member of $B$. A score is computed for each GO term considered. To test whether any of the GO terms are significantly associated with the class distinction, the null distribution is generated by permutation. For example, 10,000 GO terms can be randomly generated from the ordered genes (see Figure 2.2). For each GO term, we record the maximum ES. These measurements are considered the distribution of the null hypothesis. A $p$-value, thus, can be computed to show whether each particular gene set is significant for our null
hypothesis. The null distribution can also be generated by permuting the class label of $y$.

### 2.4.3 Parametric Analysis of Gene Set Enrichment

It has been pointed out that “GSEA is more useful when gene expression changes in a given microarray data set are minimal or moderate” (Kim and Volsky, 2005). Kim and Volsky (2005) then proceeded to propose parametric analysis of gene set enrichment (PAGE) that is proved to detect a larger number of significantly altered gene sets and their p-values were lower than the corresponding p-values calculated by GSEA. PAGE simply obtains a type of
statistic for each gene related to the GO term, and then averages them. This statistic can be any of those that represent the expression level changes across the samples, for example, fold changes, mean difference or t statistic. (Kim and Volsky (2005) use fold change as an illustrative example.) Although the distribution of this statistic of each gene may be unknown, the distribution of its average is approximately normal by the central limit theorem. Hence, Kim and Volsky (2005) use a normal null distribution to test for the significance level of each GO term.

The drawback of PAGE is that the number of genes within each GO term can not be too small. In the paper, it is suggested to use a minimal sample size of 10. Despite this sample size limit, PAGE is able to increase the power of the test from regular GSEA by comparing the average mean of the actual gene expression instead of only the ranking of the genes. Additionally, PAGE avoids finding a null distribution using permutation, like GSEA, by using a standard normal null distribution, which makes computation much less intensive. (The DEA-Mean method described later is just a special case of the PAGE where the statistic used for each gene is the mean difference.)

2.4.4 Meta-Analyses

Delongchamp et al. (2006) proposed a Meta-analysis method to compute
the overall statistical significance of a group of genes. Their method combines p-values of each gene into an overall significance level, which is adjusted for correlations among the genes.

Delongchamp et al. (2006) assume the p-values for a corresponding statistic is a random variable with a uniform distribution under true null hypothesis and it can be transformed to a convenient probability distribution. They use the inverse standard normal distribution. $z_i = \Phi^{-1}(1-p_i)$ is a random variable from the standard normal distribution and the set of p-values, \{p_i : i = 1, \cdots, m\} are from the gene level analysis for m genes related to a GO term. Assuming $p_i$s are independent,

$$\sum_{i=1}^{m} \frac{z_i}{\sqrt{m}} = \frac{1'}{\sqrt{m}},$$

also has a standard normal distribution. So the p-value,

$$P = 1 - \Phi \left( \sum_{i=1}^{m} \frac{z_i}{\sqrt{m}} \right) \quad (2.22)$$

gives an overall significance level of the gene set.

Equation 2.22 gives a naive estimate since it assumes independence of $z_i$,

where the covariance of $z_i$ is identity matrix, cov($z$) = $I$. Suppose cov($z$) = $R$, 

54
then the variance of $1'z$ is $1'R1$ and the appropriate p-value is

$$P = 1 - \Phi \left( \frac{1'z}{\sqrt{1'R1}} \right),$$

(2.23)

where $R$ is estimated from the data.

The meta-analyses proposed here combine the p-values of genes within each GO term and create an overall significance level. The overall significance level is adjusted by the correlation among the genes. Delongchamp et al. (2006) use simulation studies to show that without the adjustment for correlation, the statistics overestimate the significance when the correlations are positive.

### 2.4.5 Empirical Bayesian Generalized Linear Model

Goeman et al. (2004) proposed a global test for a group of genes, based on the empirical Bayesian generalized linear model. Goeman et al. (2004) pointed out that if a group of genes can be used to predict the clinical outcome, the gene expression patterns must differ for different clinical outcomes. This connection is used to derive the test.

Let $y$ be the clinical outcome, an $n \times 1$ vector of $n_1$ 1s and $n_0$ 0s. $X = \{x_{ji}\}$ as the $(n_0 + n_1) \times m$ data matrix containing expression levels for $m$ genes.
Model these variables with a generalized linear model as

\[ E(Y_j|\beta) = h^{-1}(\alpha + \sum_{i=1}^{m} x_{ji}\beta_i) \]  \hspace{1cm} (2.24)

where \( h \) is a link function (e.g. the logit function) and \( \beta_i \) is the regression coefficient for gene \( i, i = 1, ..., m \). Testing whether there is a predictive effect of the gene expression on the clinical outcome is equivalent to testing the hypothesis

\[ H_0 : \beta_1 = \beta_2 = \cdots = \beta_m = 0, \]

that all regression coefficients are zero. However, \( m \) might be too large relative to \( n_0 + n_1 \) and there are too few degrees of freedom for the test. Goeman et al. (2004) therefore suggested assuming that \( \beta_1, \cdots, \beta_m \) are samples from a common distribution with expectation zero and variance \( \tau^2 \). Then a single unknown parameter \( \tau^2 \) determines how much the regression coefficients are allowed to deviate from zero. The null hypothesis becomes simply

\[ H_0 : \tau^2 = 0. \]

The distribution of \( \beta \) in Equation 2.24 can be seen as a prior, with unknown shape and with a variance depending on unknown parameters. This model hence can be viewed as an empirical Bayesian model. A score test was proposed
to test for $H_0 : \tau^2 = 0$. Under such setup, the test statistic of this score test is asymptotically normally distributed. For large sample sizes, hence Goeman et al. (2004) suggested using a normal null distribution. As for small sample sizes, permutation is suggested to create the null distribution.

The test proposed by Goeman et al. (2004) can be applied to gene sets of any size. It has many nice properties of a score test. One interesting property of a score test in general is that it maximizes the average power against all alternatives where the true values of the parameters are small (Goeman et al., 2004). Therefore, this test shall have a good power under the situation where the gene set contains many genes with small signals. The test statistic showed promising performance on a set of experiment data (Golub et al., 1999), but was not compared with other standard methods such as Fisher’s exact test or GSEA. However, it should have a better power in general since the test statistic is built upon the actual gene expression levels of a gene set.

2.4.6 Stratified Mixture Model

Pan (2005) proposed a stratified mixture model to incorporate information from the Gene Ontology with the Mixture Model Methods (MMM) (Pan et al., 2001). This approach also utilizes the biological information from the Gene Ontology but in a sense to help detect differentially expressed gene better
instead of detecting significant GO terms. Recall in the previous section, MMM is used to test for individually differentially expressed genes. Assuming we have $Z_i$ as test statistic for gene $i$ to test a null hypothesis $H_{0i}$ against an alternative $H_{1i}$ for $i = 1, \cdots, M$. The distribution of $Z_i$ is modeled as a mixture of two sub-populations:

$$f(Z_i) = p_0 f_0(Z_i) + p_1 f_1(Z_i),$$

where $f_0$ corresponds to the distribution of $Z_i$ for genes when $H_{0i}$ holds and $f_1$ for that of genes when $H_{1i}$ holds; and $p_0 = Pr(H_{0i})$ and $p_1 = Pr(H_{1i})$. By Bayes theorem, the posterior probability of $H_{1i}$ is:

$$Pr(H_{1i}|Z_i) = 1 - \frac{p_0 f_0(Z_i)}{f(Z_i)}.$$

For the regular mixture model, a key assumption is that all the genes share an equal prior probability $p_0, p_1$ and the same distribution $f_0, f_1$.

Now we know that the genes are annotated in $K > 1$ GO categories, $G_1, \cdots, G_K$. Pan (2005) proposed to assign different distributions to the different $G_k$ leading to a stratified mixture model, For gene $i$ in $G_k$, $k = 1, \cdots, K$.

$$f^{(k)}(Z_i) = p_0^{(k)} f_0(Z_i) + p_1^{(k)} f_1^{(k)}(Z_i).$$

When $K = 1$, the stratified model reduces to the standard mixture model.
In general, separate mixture models are applied to various GO categories independently to find the differentially expressed genes. Pan (2006) extended the stratified mixture model to hierarchical mixture model to incorporate the hierarchical structure of the Gene Ontology. As mentioned earlier, these two extensions of the mixture model are not trying to find differentially GO terms, but to improve detecting differentially expressed genes. Nonetheless, it is an interesting but different way of using the Gene Ontology for microarray data.

2.4.7 Summaries

The development of gene annotation enhanced analysis of microarray data only started a few years ago, and it is getting more attention recently. With so much biological knowledge of genes available, how to incorporate them into analyzing microarray data provides many opportunities and challenges. Pan (2005) and Pan (2006) tried to use the biological knowledge to improve detecting differentially expressed genes, while many others have been focusing on how to detect significant biological pathways (GO terms) instead of genes to enhance interpretation in analysis of microarray data.

For procedures to detect significant GO terms, there are mainly two approaches. One approach is to build tests based on gene level test results, working with the number of differentially expressed genes (Fisher’s exact test),
gene rank of importance (GSEA) or p-values of the genes (meta-analyses). The other approach is to create a test based on the actual gene expressions related to each GO term directly, such as PAGE, empirical Bayes generalized linear model. In general, the latter approaches have better power, because the former approaches tend to lose some information along the way of gene level analysis. Our domain enhanced analysis is similar to the latter approaches, creating a test procedure that is based on the actual gene expression of each GO term. Therefore, it is expected to have good power for detecting important GO terms.
Chapter 3

Domain Enhanced Analysis

3.1 Introduction

Advances in microarray technology have greatly enhanced gene expression studies. In these studies, thousands of genes are monitored and probed on only a few to tens of samples. Gene expression data present both great opportunities and challenges. Patterns of gene expression can be used to determine genes with similar behavior, suggest biomarkers for a specific disease, and propose targets for drug intervention. However, several aspects of gene expression data analysis make it difficult to apply classical statistical methods:
• Gene expression data all share the common problem of $p \gg n$, where $p$ is the number of genes and $n$ is the number of samples. A typical microarray data set has thousands of genes, but often less than 100 samples.

• Many of the genes are involved in multiple biological pathways and are therefore highly correlated.

• Interpretation of analysis output is problematic when hundreds of genes are identified as important.

Various dimension reduction approaches have been developed to alleviate the problem of $p \gg n$. For example, Li and Li (2004) proposed a dimension reduction method by combining principal components analysis (PCA) and sliced inverse regression (SIR) to produce linear combinations of genes that capture both the underlying variation of gene expressions and the phenotypic information, and then use the extracted combination of genes in the subsequent survival model formulation.

Our goal, however is not only to solve the common $p \gg n$ problem of high dimensional data, but also to enhance the interpretability of the models within the context of biological knowledge. While Li and Li (2004) offer a reduced-dimension model with some added interpretability, the interpretability is driven by statistical learning, not biological context.
Others have tried to analyze gene expression data focusing on domain-aggregated results to increase interpretability. The most common methods use a standard “bottom-up” approach where genes are first tested individually and then aggregated according to biological functions or pathways by domain knowledge such as the Gene Ontology (GO) by Ashburner et al. (2000). This type of knowledge structure provides great utility in the annotation and biological interpretation of gene sets obtained in microarray experiments. Ashburner et al. (2000) enable functional annotations of a given gene set by clustering genes according to their biological characteristics. Furthermore, through the GO hierarchical structure, it is also possible to represent biological concepts with different conceptual levels, from very general to very precise.

Various software tools are currently available for the ontological analysis of high throughput gene expression experiments. GenMapp (Dahlquist et al., 2002), ChipInfo (Zhong et al., 2003), GoMiner (Zeeberg et al., 2003), GeneMerge (Castillo-Davis and Hartl, 2003), FatiGO (Al-Shahrour et al., 2003), OntoTools (Draghici et al., 2003b), FuncAssociate (Berriz et al., 2003), GOstat (Beissbarth and Speed, 2004) and ErmineJ (Lee et al., 2005a) are some of the popular ones. Khatri and Draghici (2005) did a comparative study of some commonly used tools for analyzing gene expressions in light of GO enrichment; their study was from a software point of view, and as such focused
on aspects such as scope of the reported analysis, user interface, platform, type of application (web-based or not), etc. Though most of these tools have different implementation, they use the same basic procedure. All tools start by identifying differentially expressed genes via some ordering metric. Statistical hypotheses are then created to test whether each GO term contains an unusually large number of differentially expressed genes, in which case that GO term is identified as important.

Several statistical approaches have been used to calculate p-values for each GO term. Draghici et al. (2003a) showed that the number of significant genes in a given term can be modeled by a hypergeometric distribution, and hence a test may be carried out with the distributional information. More popular approaches are the chi-square test for equality of proportions and Fisher’s exact test. To apply these two tests, a contingency table is created as shown in Table 3.1. Man et al. (2000) performed extensive simulations to show that the chi-square test has better power and robustness than Fisher’s exact test. Fisher’s exact test is usually applied only when at least one of the expected values in a contingency table is smaller than five because in this case the chi-square test is no longer appropriate.

A different approach is proposed by Mootha et al. (2003), and it is called gene set enrichment analysis (GSEA); see also Subramanian et al. (2005).
Table 3.1: Contingency table classification of genes according to whether they are identified as differentially expressed (flagged) and whether they are part of the GO term.

<table>
<thead>
<tr>
<th></th>
<th>Flagged Genes</th>
<th>Non-flagged Genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>In GO term</td>
<td>n₁₁</td>
<td>n₁₂</td>
</tr>
<tr>
<td>Not in GO term</td>
<td>n₂₁</td>
<td>n₂₂</td>
</tr>
</tbody>
</table>

Rather than performing a test based on the number of differentially expressed genes in a GO term, they first create a score for each GO term based on the rank of significance for all individual genes contained in that GO term. The resulting p-value for the score of each GO term is found by comparison to the null distribution of the scores as obtained using permutation. Generally speaking, these “bottom-up” gene set enrichment methods improve the ability to interpret results from gene level analysis. The disadvantage is that they all require gene-level analysis prior to conducting a GO-level analysis. Results from the gene-level analysis are directly related to the effectiveness of these methods. For some of these pre-processing procedures, we need to identify flagged or non-flagged genes, meaning that a statistical threshold has to be set. This threshold can have a major impact on the analysis results, but is not easy to define. Several other approaches have been proposed, including the random effects model of Goeman et al. (2004), the PAGE approach of Kim and Volsky (2005), the approach of Tian et al. (2005), the SAFE approach of Barry et al. (2005) and the composite GO annotation approach of ADGO by Nam et al. (2006).
We propose a “top-down” approach, in which we find a summary statistic for each GO term; this summary statistic can be viewed as a new latent variable created from the individual genes in that GO term. These sets of biologically related genes can create a smaller number of new variables with informative descriptions of the biology. If the ontology is accurate, we shall see most of the highly correlated genes being grouped in one set. On one hand, we can reduce the number of variables and alleviate the problem of \( p \gg n \). On the other hand, we are able to increase interpretability of our findings like the other enrichment methods. These new meaningful variables can be used to either test for significance of each GO term or make predictive models.

In this paper, we focus on the testing aspect of our methods. We introduce and explain our procedure in the Methods section. In the Results section, we use both simulated data and results from actual experiments to demonstrate our procedure’s ability to identify significant GO terms. Our method is also compared to other approaches. We discuss advantages and potential drawbacks of our method in the last section.
3.2 Methods

Our proposed method is called domain enhanced analysis (DEA). It is a “top-down” approach that aggregates the genes before we proceed to the analysis. The procedure is described below:

1. Group genes into each GO term. (Some genes will exist in multiple terms.)

2. Create a new latent variable for each GO term by combining the individual genes within that GO term.

3. Use the latent variable to test the predictive ability of each GO term.

4. Adjust p-values accordingly for multiple testing.

Figure 3.1 is a flow chart to illustrate our method. The procedure is straightforward but has much room for expansion according to choices made for steps 2, 3 and 4. The simplest choice for step 2 is to use an unweighted average as the latent variable; we call this approach DEA-Mean. We also investigated aggregation based on the first principal component (DEA-PCA) and on the first latent variable (score) from partial least squares (DEA-PLS) by Hoskuldson (1988) or de Jong (1993). We use the “PLS” procedure in SAS to find PLS
latent variables and the “PRINCOMP” procedure in SAS to find PCA latent variables.

![Flow chart illustrating the procedure](image)

**Figure 3.1:** A flow chart illustrates our procedure. A: steps 1 and 2 aggregate genes and create latent variables. For PLS procedure, treatments are considered as response variable. B: steps 3 and 4 test for significance of each latent variable with respect to treatment and adjust p-values for multiple testing.

Consider the set of genes for a GO term as forming a predictor matrix $X$ of $n$ rows (samples) and $p$ columns (number of genes in the set). A corresponding $n \times q$ matrix $Y$ may represent $q$ responses or $q$ treatment factors for each sample; in either case we refer to $Y$ as the response. PLS finds a linear combination of the columns of $X$ that has maximum covariance with $Y$. PCA finds a linear combination of the columns of $X$ to maximize the variance of $X$ without regard
to $Y$, and the mean is not designed to maximize any particular variation. Based
on preliminary simulation results (not presented here), PLS is expected to out-
perform the other two options.

For the applications in this paper, the “response” is actually univariate
and indicates membership in one of two classes. As such, $y$ represents the
$n$-vector of responses and we choose to use a 0/1 coding for these responses.
Preliminary theoretical derivations show that both the 0/1 and $-1/1$ codings
for $y$ magnify effects of differentially expressed genes, but these codings do not
result in equivalent test statistics or properties. This work is forthcoming in a
separate paper. Furthermore, our DEA procedure can be expanded to three or
more classes. With either two binary vectors or one multi-class vector, the PLS
procedure can be applied to find the first latent variable of the gene expressions.
An ANOVA or F test can be applied instead of the t-test to test for predictive
ability of each latent variable with respect to a multi-class response.

The PLS algorithm used in the paper is by de Jong (1993). It is designed for
continuous response variables and we use it without modification even though
in our case the response is a dichotomous quantity. The problem of using
a categorical response in PLS has been considered by Marx (1996), Nguyen
and Rocke (2002a,b, 2004), Huang and Pan (2003), Bastien et al. (2005), Ding
and Gentleman (2005) and Fort and Lambert-Lacroix (2005). These alternative
algorithms can potentially improve the summary latent variables for DEA-PLS. Others extension of PLS, such as the nonlinear partial least squares (Malthouse et al., 1997), can also be worthy of consideration for special cases.

Because PLS finds a linear combination of $X$ by looking at the relationship between $X$ and $y$, it is subject to random variation. To overcome this, cross validation is used to determine whether the first PLS latent variable significantly contributes to the prediction of $y$. After evaluating predicted residual sum of squares (PRESS) for a model where no PLS latent variables are retained (so that prediction occurs using the mean response) and a model that performs PLS-regression using only one retained PLS latent variable, the test procedure by van der Voet (1994) is applied to detect significant improvement from the one-latent-variable model. If the first PLS latent variable for a GO term overfits $y$, we consider that gene set as non-important and simply omit it from steps 3 and 4 above. An alternative to completely dropping this gene set would be to assign it a very large p-value so that it gets very little attention in steps 3 and 4. By filtering gene sets in this manner, we limit the chances of retaining irrelevant latent summaries. As expected, we find that filtering is most intense for GO terms that contain large numbers of genes. It is likely that these cases require more than one PLS summary variable, but for now we limit ourselves to retaining at most one PLS latent variable; extension to
retaining multiple PLS summaries is certainly possible. Cross validation and van der Voet’s test are implemented in SAS PLS using options “CVTEST” and “CV=SPLIT” with a default significance level of 0.10. (See Appendix A for the SAS codes.)

For the testing procedure in step 3, we use the equal variance two sample t-test throughout the paper. Other choices of tests are mentioned in the Discussion section. Adjustment for multiplicity in step 4 is done using the approach of Benjamini and Hochberg (1995) to control the false discovery rate (FDR); we will refer to this as the BH adjustment.

### 3.3 Results

#### 3.3.1 Simulation Study

To investigate the properties of our proposed DEA method, we carry out several simulated studies. The focus of our simulation is to verify the effectiveness of our PLS summary measurement. We also compare DEA-PLS to DEA-Mean, the Fisher’s exact approach and GSEA by Mootha et al. (2003).

For the simulated data set, we inherit the mapping structure between GO and genes from a real data set with 3666 genes. These genes are mapped to
556 GO terms from the molecular function hierarchy. Instead of using the expression levels from the experimental data, we simulate them so that we know what GO terms are supposed to be important. After selecting our binary response variable $y$, we generate differentially and non-differentially expressed genes from a conditional normal distribution to form matrix $X$.

We start by setting $y$ to be either 1 or 0 by using a Bernoulli(0.5) distribution. The response $y$ is set to be 1s and 0s throughout the paper. Other choices of response variable $y$ are mentioned in the Methods section. The sample size is set to 100. Each gene expression $x$ is generated as: $x|y = 0 \sim N(-\mu, 1)$, or $x|y = 1 \sim N(\mu, 1)$. For non-important genes, $\mu$ is set to zero, while for important genes, we set $\mu$ to be some value depending on the extent of significance of each particular gene. In other words, differentially expressed genes are distributed as a mixture of two normals with common variance.

For this simulated study, there are 16 differentially expressed genes out of a total of 3666 genes: $\mu = \delta$ for genes 1514 to 1522 of $X$; $\mu = 1.2\delta$ for gene 2002; $\mu = 1.3\delta$ for genes 2872, 2874, and 2887 to 2889; $\mu = 1.8\delta$ for gene 1283; and $\delta$ takes values 0, 0.1, 0.3, 0.5, 0.7, 0.9. These differentially expressed genes are associated with nine molecular functions, as detailed in Table 3.2. Molecular functions 139, 384, 415, 601, and 725 are annotated only to differentially expressed genes, while molecular functions 142, 622, 763, and 931 are also an-
notated to some genes that are not simulated to be differentially expressed. We refer to the former as fully-important molecular functions and to the latter as partially-important molecular functions. The degree of partial importance for the latter group of molecular functions can be quite small as seen in Table 3.2. For example, molecular function 622 has only one differentially expressed gene but 415 non-differentially expressed genes.

Table 3.2: Gene counts within fully or partially important molecular functions.

<table>
<thead>
<tr>
<th>MFs</th>
<th>Non-important Genes Counts</th>
<th>Important Gene Counts</th>
</tr>
</thead>
<tbody>
<tr>
<td>MF139</td>
<td>0</td>
<td>10</td>
</tr>
<tr>
<td>MF384</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>MF415</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>MF601</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>MF725</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>MF763</td>
<td>415</td>
<td>1</td>
</tr>
<tr>
<td>MF931</td>
<td>14</td>
<td>1</td>
</tr>
</tbody>
</table>

We first create a single simulation replicate generated using $\delta = 0.3$. An equal-variance two-sample t-test is conducted for each gene, where samples are defined according to $y = 0$ or 1. Adjustment for multiplicity is done using the BH procedure. After adjustment, none of the genes are detected to be important using $\alpha_I = 0.05$. Hence, if Fisher’s exact test were used to test for significant GO terms, none would be found simply because no signal would be detected at the gene level. In other words, Fisher’s exact test approach fails to identify any molecular function as being important since all Fisher’s exact p-values equal one. It is possible to get around this by changing the gene-level threshold $\alpha_I$, but finding a justifiable threshold can be problematic. Another
option is GSEA. Since GSEA creates scores for each GO term using the gene ranking of importance, it avoids the problem of specifying $\alpha_I$.

Table 3.3 provides results from GO-level analyses for the single simulation replicate we generated using $\delta = 0.3$. After obtaining either the PLS or mean summary for a GO term, an equal-variance two sample t-test is conducted, where samples are again defined according to $y = 0$ or 1. For GSEA, we create scores for each GO term and test each score using a null distribution generated by permutation described by Mootha et al. (2003). The size of the permutation is 10,000. BH-adjusted p-values for these methods are presented in Table 3.3 for all nine important molecular functions. Using any reasonable $\alpha_G$ GO-level threshold for these multiplicity-adjusted p-values, it is clear that DEA-PLS and DEA-Mean outperform GSEA. By only looking at the rank-ordering of the genes, GSEA loses power by ignoring the actual gene expression level.

Table 3.3: BH-adjusted p-values of important GO terms from different methods. These results are from a single simulation replicate where $\delta = 0.3$. P-values for Fisher’s exact test equal 1 in every case.

<table>
<thead>
<tr>
<th>MFs</th>
<th>p-values for DEA-PLS</th>
<th>DEA-Mean</th>
<th>GSEA</th>
</tr>
</thead>
<tbody>
<tr>
<td>MF139</td>
<td>$2.92 \times 10^{-3}$</td>
<td>$1.43 \times 10^{-15}$</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>MF415</td>
<td>$8.65 \times 10^{-4}$</td>
<td>$8.65 \times 10^{-17}$</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>MF601</td>
<td>$2.58 \times 10^{-15}$</td>
<td>$4.81 \times 10^{-13}$</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>MF142</td>
<td>$2.98 \times 10^{-4}$</td>
<td>$4.43 \times 10^{-8}$</td>
<td>&lt; 0.0001</td>
</tr>
<tr>
<td>MF763</td>
<td>$6.19 \times 10^{-8}$</td>
<td>$6.51 \times 10^{-1}$</td>
<td></td>
</tr>
<tr>
<td>MF725</td>
<td>$1.03 \times 10^{-8}$</td>
<td>$2.07 \times 10^{-8}$</td>
<td>$7.37 \times 10^{-4}$</td>
</tr>
<tr>
<td>MF931</td>
<td>* Non-Significant $3.87 \times 10^{-1}$</td>
<td>$8.51 \times 10^{-1}$</td>
<td></td>
</tr>
<tr>
<td>MF622</td>
<td>* Non-Significant $8.30 \times 10^{-1}$</td>
<td>$8.51 \times 10^{-1}$</td>
<td></td>
</tr>
</tbody>
</table>

* Cross-validated PLS resulted in no GO-term summary variable.

DEA-PLS has the smallest p-values and finds six of the nine important GO
terms with no false discovery. Two of the missed GO terms have marginal
signals; “MF622” and “MF931” both only relate to one important gene and
many unimportant genes. It is more likely that we should regard them as non-
important GO terms. It is interesting to note that “MF384” and “MF415”
are listed as important and non-important respectively. Both have only one
gene mapped to them, and they are both simulated to be important genes. At
gene-level analysis, both genes are listed as non-significant, while at GO-level
analysis, one is significant and the other is not. It is because at the gene level
we are doing 3666 t-tests, but at the GO level, we are only doing 556 t-tests.
With BH adjustment, it turns out that the threshold of significance is just
between the two GO terms. In other words, by decreasing the number of tests,
we are able to increase power even for those GO terms mapped to only one
gene.

Previous results were limited to a single simulation replicate to allow the
reader a detailed comparison of the Fisher’s exact approach, GSEA, DEA-PLS
and DEA-Mean. We now move to 50 simulation replicates to provide a more
comprehensive comparison. Sensitivities and specificities of the four methods
are shown in Figure 3.2. For these results, $\alpha_I = 0.05$ and $\alpha_G = 0.05$.

By looking at multiple runs of the simulated data sets, it is interesting to
note that Fisher’s exact test performs reasonably well for $\delta$ sufficiently large.
Figure 3.2: Sensitivity and specificity of four methods as a function of $\delta$ in the simulation study. Sensitivities are based on averages across all simulation replicates, with pointwise approximate 95% confidence intervals shown as vertical bars. Fisher’s exact test is solid line, DEA-PLS is dashed line, DEA-Mean is dotted line and GSEA is dashed-dotted line. $\alpha_I = 0.05$ for Fisher’s exact test. All GO-level p-values were BH-adjusted using $\alpha_G = 0.05$.

Fisher’s exact test has an average sensitivity of 54% when $\delta = 0.3$ and 77% when $\delta = 0.9$, which is much better than GSEA’s 34% when $\delta = 0.9$. It is quite clear, however, that DEA-PLS is able to detect more true findings than the other three methods. It detects 20% more signals than Fisher’s exact test and 60% more than GSEA. On the other hand, DEA-Mean only has marginally better sensitivity than Fisher’s exact test. The improved sensitivity of DEA-PLS does not come at the cost of relevant loss of specificity. As seen in Figure 3.2, all methods have approximately equal levels of specificity. Hence we can say that DEA-PLS is able to increase sensitivity of the analysis without having too
many false discoveries.

We also use our simulation study to investigate the effect of \( \alpha_I \) on Fisher’s exact test. For 50 simulation replicates with \( \delta = 0.3 \), we apply Fisher’s exact test repeatedly for \( \alpha_I = 0, 0.01, 0.03, 0.05, 0.07, 0.10, 0.20, 0.50, 0.70 \). The resulting ROC curve is presented in Figure 3.3. Clearly, sensitivity can change quite drastically as \( \alpha_I \) changes. This shows that the effectiveness of Fisher’s exact test is directly related to whether we choose the ideal \( \alpha_I \) threshold for gene-level analysis. We also plot (sensitivity, 1-specificity) pairs for DEA-PLS, DEA-Mean and GSEA in the graph. DEA-PLS has a better sensitivity overall than Fisher’s exact test at its best, while DEA-Mean performs almost equivalent to Fisher’s exact test for one value of \( \alpha_I \). In our studies, GSEA has the worst sensitivity.

With the simulation studies, we can see that the drawback of most of the existing methods is that they require gene-level analysis, either finding the important genes or the rank of significance of the individual genes. For situations where gene-level signals are weak, these methods perform poorly, because their effectiveness relies largely on strong signals at the gene level. On the contrary, DEA methods do not require strong gene-level signals or gene-level analysis. Furthermore, they are able to combine the weak signals for genes related to each GO term and make them easier to detect. The simulation results show
Figure 3.3: ROC curve for Fisher’s exact test by changing the threshold $\alpha_I$ for gene-level analysis. Data was generated following the simulation design using $\delta = 0.3$. Estimates are based on 50 simulation replicates. Vertical bars are pointwise approximate 95% confidence interval. Also shown are points for DEA-PLS, DEA-Mean, and GSEA; These are single points because they do not depend on $\alpha_I$. 
DEA methods are effective for finding GO-level signals. Fisher’s exact test performs reasonably well when gene level analysis does work, but the $\alpha_I$ threshold for gene-level analysis is critical for the effectiveness of Fisher’s exact test. On the other hand, GSEA has the weakest power to detect signals at the GO level. For all simulated data sets, DEA-PLS has the best power to detect significant GO terms.

### 3.3.2 Experimental Data Study

We evaluate DEA-Mean and DEA-PLS against the classic Fisher’s exact test on two real gene expression data sets by Chiaretti et al. (2004) and Golub et al. (1999). Both data were collected to identify genes that distinguish subgroups of leukemia patients. The first data set splits patients into B-cell and T-cell type acute lymphoblastic leukemia (ALL), while the second data set consists of patients with ALL and acute myeloid leukemia (AML). Both data sets have been extensively studied in the literature of microarray analysis and are available in R as Bioconductor packages, “ALL” and “golubEsets” (http://www.bioconductor.org/). We present the analysis results of Chiaretti’s data set and results of Golub’s data set in this section.
Leukemia data set by Chiaretti

The data set consists of 128 patients with ALL. It is known that ALL cells are delivered from either B-cell or T-cell precursors. Among these patients, there are 95 with B-cell ALL and 33 with T-cell ALL. The HGU95aV2 Affymetrix chip was used for the experiment. We annotate the data using GO’s biological process (BP) ontology. 10012 probe sets are annotated from a total of 12625 probe sets in the data. The annotation includes 1955 GO term nodes. These GO terms all are mapped to at least one probe ID in the data. (See Appendix C for codes for generating mappings from genes to GO terms using bio-conductor.)

In some literature, mappings are first done from probe sets to EntrezGene ID and then to the GO terms. In our implementation, we simply use the mapping directly from probe sets to GO terms to be consistent with DEA, which finds a summary for each GO term using all gene expressions in the GO term. We are careful not to add a step of mapping from probe sets to EntrezGene ID and thus weaken the comparison between DEA and Fisher’s exact test. To keep our analysis consistent, for our implementation of Fisher’s exact test, we also use the mapping from probe sets directly to GO terms.

To apply Fisher’s exact test, we first perform a gene-level analysis on
each probe to find differentially expressed probe sets. Functions “lmFit” and “eBayes” in R package “limma” are used to fit a linear model for each probe set and produce a p-value for each one. The p-values are BH-adjusted. Using level $\alpha_I = 0.01$, a total of 2016 probe sets are found to be significant. We then proceed to apply one-sided Fisher’s exact tests using results from the gene-level analysis. We also apply DEA-PLS and DEA-Mean methods. All three methods find many signals at the GO term level. We can possibly reduce the number of significant findings by adjusting our t-test, but it is not the focus of this paper. For now, we suggest focusing on the most significant GO terms.

In Table 3.4, we present statistics for the ten most significant GO terms detected by Fisher’s exact test. They mostly consist of biological processes related to antigen processing and presentation. These findings are mentioned in previous biological literature including Novak (2001), Dalla-Favera (2001) and Look (2001). DEA-PLS is also able to detect these signals and the p-values from DEA-PLS are much smaller. DEA-PLS using all the information from the probe level has greater power than Fisher’s exact test because information is lost when the actual expression level of each probe is ignored. Table 3.4 lists the p-values of each GO term followed by the rank of that GO term by both methods. Three of the GO terms are consistently being selected by DEA-PLS and Fisher’s exact test among their ten most significant GO terms: immune
response, defense response and response to biotic stimulus. Excluding these three GO terms, the most significant GO terms identified by Fisher’s exact test are not the most significant ones identified by DEA-PLS.

In Table 3.5, the ten most significant GO terms from DEA-PLS are listed. DEA-PLS assigns greater significance to GO terms related to cell activity than to those related to antigen processing and presentation. Specifically, DEA-PLS has smaller p-values for T-cell selection, immune cell activation and positive regulation of T cell receptor signaling pathway. From the literature, distinct gene expression signatures related to cell functions were recognized by Novak (2001), Dalla-Favera (2001) and Look (2001). It also makes perfect sense that the best GO terms to differentiate T-cell and B-cell ALL would be those probes related to cell functions. From the results, we can see that Fisher’s exact test detected some of these signals in a less significant role, for example, immune cell activation, cell activation and lymphocyte activation. For some other GO terms like T-cell selection and positive regulation of T-cell receptor signaling pathway, Fisher’s exact test cannot effectively identify signals. At this point, we would like to know whether it is justifiable to claim that GO terms found by DEA-PLS are more relevant than those found by Fisher’s exact test.

Without enough comparative studies in the literature on the extent of importance for these GO terms differentiating leukemia subtypes, we try to find
the answer by drilling down to the probe level. Searching through probe-level analysis, we find that four of the five most important probes are clearly related to the most significant GO terms found by DEA-PLS. More specifically, lymphocyte differentiation, lymphocyte activation, immune cell activation and cell activation are related to the first, third, fourth, fifth and ninth of the ten most significant probe sets. T-cell selection is one of the more precise terms. With only eight probe sets mapped to it, first and fourth most significant probe sets are among them. The most significant GO terms found by DEA-PLS are mapped to some of the most significant probe sets. Likewise, these probes were also significant using Fisher’s exact test. The sixth, seventh, eighth and tenth most significant probe set are mostly related to antigen processing and presentation. DEA-PLS also finds these GO terms to be significant, but only in a slightly less significant role.

Table 3.4: GO-level analysis results for ten most significant GO terms found by Fisher’s exact test in Chiaretti’s data.

<table>
<thead>
<tr>
<th>GO Term</th>
<th>O</th>
<th>A</th>
<th>Fisher’s Exact</th>
<th>DEA-PLS (Rank)</th>
</tr>
</thead>
<tbody>
<tr>
<td>antigen presentation</td>
<td>40</td>
<td>59</td>
<td>1.30 x 10^{-15}</td>
<td>2.02 x 10^{-38} (89)</td>
</tr>
<tr>
<td>antigen presentation, exogenous antigen</td>
<td>17</td>
<td>18</td>
<td>4.04 x 10^{-10}</td>
<td>2.07 x 10^{-35} (134)</td>
</tr>
<tr>
<td>antigen processing</td>
<td>35</td>
<td>55</td>
<td>4.82 x 10^{-10}</td>
<td>7.20 x 10^{-37} (110)</td>
</tr>
<tr>
<td>antigen presentation, endogenous antigen</td>
<td>30</td>
<td>45</td>
<td>1.62 x 10^{-9}</td>
<td>1.23 x 10^{-36} (115)</td>
</tr>
<tr>
<td>antigen processing, endogenous antigen via MHC class I</td>
<td>29</td>
<td>45</td>
<td>9.87 x 10^{-9}</td>
<td>1.55 x 10^{-34} (155)</td>
</tr>
<tr>
<td>antigen processing, endogenous antigen via MHC class II</td>
<td>17</td>
<td>20</td>
<td>1.53 x 10^{-8}</td>
<td>1.98 x 10^{-35} (132)</td>
</tr>
<tr>
<td>defense response</td>
<td>283</td>
<td>913</td>
<td>3.16 x 10^{-7}</td>
<td>1.02 x 10^{-53} (6)</td>
</tr>
<tr>
<td>immune response</td>
<td>259</td>
<td>831</td>
<td>6.46 x 10^{-7}</td>
<td>1.93 x 10^{-54} (5)</td>
</tr>
<tr>
<td>response to biotic stimulus</td>
<td>292</td>
<td>955</td>
<td>7.79 x 10^{-7}</td>
<td>2.05 x 10^{-53} (8)</td>
</tr>
<tr>
<td>detection of pest, pathogen or parasite</td>
<td>11</td>
<td>13</td>
<td>7.28 x 10^{-6}</td>
<td>1.07 x 10^{-34} (131)</td>
</tr>
</tbody>
</table>

A – the number of probe sets annotated for each GO term.
O – the number of significant probe sets annotated for each GO term.
Table 3.5: GO-level analysis results for ten most significant GO terms found by DEA-PLS in Chiaretti’s data.

<table>
<thead>
<tr>
<th>GO Term</th>
<th>O</th>
<th>A</th>
<th>Fisher’s Exact (Rank)</th>
<th>DEA-PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>T cell selection</td>
<td>5</td>
<td>8</td>
<td>2.31 x 10^{-2} (122)</td>
<td>5.96 x 10^{-5}</td>
</tr>
<tr>
<td>immune cell activation</td>
<td>47</td>
<td>114</td>
<td>3.47 x 10^{-5} (12)</td>
<td>9.87 x 10^{-56}</td>
</tr>
<tr>
<td>cell activation</td>
<td>47</td>
<td>115</td>
<td>4.52 x 10^{-5} (13)</td>
<td>9.87 x 10^{-56}</td>
</tr>
<tr>
<td>transition metal ion homeostasis</td>
<td>9</td>
<td>28</td>
<td>2.13 x 10^{-1} (463)</td>
<td>1.24 x 10^{-55}</td>
</tr>
<tr>
<td>immune response</td>
<td>259</td>
<td>831</td>
<td>6.46 x 10^{-7} (8)</td>
<td>1.93 x 10^{-54}</td>
</tr>
<tr>
<td>defense response</td>
<td>283</td>
<td>913</td>
<td>3.16 x 10^{-7} (7)</td>
<td>1.02 x 10^{-53}</td>
</tr>
<tr>
<td>lymphocyte activation</td>
<td>40</td>
<td>101</td>
<td>3.56 x 10^{-4} (22)</td>
<td>2.05 x 10^{-53}</td>
</tr>
<tr>
<td>response to biotic stimulus</td>
<td>292</td>
<td>955</td>
<td>7.79 x 10^{-7} (9)</td>
<td>2.05 x 10^{-53}</td>
</tr>
<tr>
<td>hemopoiesis</td>
<td>28</td>
<td>94</td>
<td>1.19 x 10^{-1} (314)</td>
<td>2.08 x 10^{-52}</td>
</tr>
<tr>
<td>positive regulation of T cell receptor signaling pathway</td>
<td>3</td>
<td>4</td>
<td>4.56 x 10^{-2} (165)</td>
<td>5.60 x 10^{-52}</td>
</tr>
</tbody>
</table>

A – the number of probe sets annotated for each GO term.
O – the number of significant probe sets annotated for each GO term.

Some of the advantages of using GO to annotate genes are its hierarchical structure and many available tools for displaying GO terms. The Gene Ontology web site http://www.geneontology.org/ lists many useful tools for searching and browsing GO terms. We use a web-based tool “AmiGO” available at http://www.godatabase.org/cgi-bin/amigo/go.cgi to display our results from DEA-PLS and Fisher’s exact test. The output graphic for Fisher’s exact test is presented in Figure 3.4 and for DEA-PLS it is displayed in Figure 3.5. In each figure, the ten most significant GO terms are underlined. Their rank of importance for each respective method is at the upper left corner of each node, while the rank of importance of the most significant probe sets contained in the GO term are shown in the curly bracket. By displaying the hierarchical structures of GO terms, we are able to gain additional insight. If we only look at the ten most significant GO terms found by both methods, these terms are
made up of two distinct subtrees. First of all, these two subtrees are both descendants of the GO terms immune response, defense response and response to biotic stimulus, which are nested by sequence. This explains why these three GO terms are identified as significant by both DEA-PLS and Fisher’s exact test. Furthermore, the important GO terms found by both methods are nested nicely to each other, which shows how these biological functions are being affected by different leukemia subtypes from more precise terms to more general terms. It is encouraging to see that the subtree of GO terms found by DEA-PLS is related to more significant probe sets than those terms found by Fisher’s exact test, suggesting that DEA-PLS is maintaining the right order of importance for the GO terms found.

In conclusion, it is clear that both DEA-PLS and Fisher’s exact test are able to pick out some useful significant GO terms. DEA-PLS has better power with all the information included, and hence it is able to find more signals at the GO term level. Furthermore, DEA-PLS is able to find the most significant GO terms that are related to the most significant probe sets. By combining the actual gene expression information for each GO term, DEA-PLS is able to maintain the right order of importance for the GO terms. This simply cannot be achieved by Fisher’s exact test because it treats all the important probe sets with equal weight. With four of the top five most significant probe sets related
Figure 3.4: GO graph of findings by Fisher's exact test for Chiaretti's Data. Specific terms are at the top, while general terms are at the bottom. Top ten significant GO terms are underlined, with rank shown in upper left of rectangle. Ranks for top ten probe sets are shown in curly brackets.

to biological processes like immune cell activation, lymphocyte activation and lymphocyte differentiation etc, they certainly should be and are able to stand out from the rest by using DEA-PLS.

We also carried out DEA-Mean in the analysis, though we did not list the results here. DEA-Mean is straightforward and fast to implement, but we find that its behavior can be erratic. One obvious drawback is that it has very weak power when the number of probes related to a GO term becomes large. Hence, DEA-Mean finds immune response, defense response and response to biotic
Figure 3.5: GO graph of findings by DEA-PLS for Chiaretti's Data. Specific terms are at the top, while general terms are at the bottom. Top ten significant GO terms are underlined, with rank shown in upper left of rectangle. Ranks for top ten probe sets are shown in curly brackets.
stimulus to be less significant while they are being recognized very significant by both DEA-PLS and Fisher’s exact test.

**Leukemia data set by Golub**

Here we take a look at another leukemia data set by Golub *et al.* (1999). The HGU6800 Affymetrix chip was used for this experiment. This data set consists of 73 bone marrow samples with either acute lymphoblastic leukemia (ALL) or acute lymphoblastic leukemia (AML). In our analysis, we use the full data set, among which 47 of the bone marrow samples are with ALL and 25 of them are with AML. There are a total of 7129 probe sets. Again, we annotate these probe sets directly with biological process ontology from GO. 6270 probe sets are mapped to 1555 biological process GO terms after annotation. Each of these GO terms is mapped to at least one probe ID.

Using Function “lmFit” and “eBayes” in R package “limma” for our probe level univariate analysis, we find 456 significant probe sets that differentiate leukemia subtypes with $\alpha_I = 0.01$ tests adjusted by FDR. One-sided Fisher’s exact test is applied using the probe level results. At the same time, we apply DEA-PLS and DEA-Mean to the gene expression data. After creating summary variable for each GO term, we apply two-sided t-test and adjust the p-values with FDR. We find large number of significant GO terms by both methods,
especially for DEA-PLS. Again, we suggest to focus on the most significant GO terms, say 150 GO terms.

We list the ten most significant GO terms found by Fisher’s exact test along with their results from DEA-PLS and DEA-Mean in Table 3.6. Except for “cell wall catabolism”, the findings by Fisher’s exact test are consistent with DEA-PLS. They are mostly among the top 100 most significant GO terms found by DEA-PLS. As for “cell wall catabolism”, it is a very precise term with only four probe sets mapped. Even though all four of these probe sets are significant, none of them are listed among the top ten most significant ones and one of these probe sets is borderline significant. Hence, DEA-PLS does not find it to be very significant compared with other GO terms. DEA-Mean does not perform very well in general since the number of probe sets mapped to these GO terms is pretty large and we already point out DEA-Mean loses power under this condition.

Table 3.6: GO-level analysis results for top 10 most significant GO terms found by Fisher’s exact test in Golub’s data.

<table>
<thead>
<tr>
<th>GO Term</th>
<th>O</th>
<th>A</th>
<th>Fisher’s Exact</th>
<th>DEA-PLS (Rank)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. response to biotic stimulus</td>
<td>84</td>
<td>751</td>
<td>2.30 × 10⁻²</td>
<td>6.04 × 10⁻¹³  (10)</td>
</tr>
<tr>
<td>2. cell wall catabolism</td>
<td>4</td>
<td>4</td>
<td>2.79 × 10⁻⁵</td>
<td>2.88 × 10⁻⁸   (628)</td>
</tr>
<tr>
<td>3. defense response</td>
<td>80</td>
<td>730</td>
<td>7.76 × 10⁻⁵</td>
<td>5.63 × 10⁻¹⁰ (4)</td>
</tr>
<tr>
<td>4. catabolism</td>
<td>42</td>
<td>383</td>
<td>3.74 × 10⁻⁴</td>
<td>8.73 × 10⁻¹⁵ (20)</td>
</tr>
<tr>
<td>5. response to stress</td>
<td>81</td>
<td>785</td>
<td>5.22 × 10⁻⁴</td>
<td>6.47 × 10⁻¹⁷ (54)</td>
</tr>
<tr>
<td>6. response to stimulus</td>
<td>130</td>
<td>1388</td>
<td>6.03 × 10⁻⁴</td>
<td>5.32 × 10⁻¹⁷ (43)</td>
</tr>
<tr>
<td>7. metabolism</td>
<td>302</td>
<td>3695</td>
<td>6.52 × 10⁻⁴</td>
<td>4.81 × 10⁻¹⁰ (105)</td>
</tr>
<tr>
<td>8. response to other organism</td>
<td>52</td>
<td>458</td>
<td>7.06 × 10⁻⁴</td>
<td>2.16 × 10⁻¹⁰ (75)</td>
</tr>
<tr>
<td>9. cellular catabolism</td>
<td>36</td>
<td>289</td>
<td>9.37 × 10⁻⁴</td>
<td>7.52 × 10⁻¹⁰ (17)</td>
</tr>
<tr>
<td>10. peptidyl-amino acid modification</td>
<td>9</td>
<td>37</td>
<td>1.06 × 10⁻³</td>
<td>1.39 × 10⁻¹⁷ (27)</td>
</tr>
</tbody>
</table>

A – the number of probe sets annotated for each GO term.
O – the number of significant probe sets annotated for each GO term.
In Table 3.7, we list the most significant GO terms found by DEA-PLS. For this data set, DEA-PLS shares quite a few common findings with Fisher’s exact test, GO terms “nucleotide metabolism”, “defense response”, “immune response”, “coenzyme metabolism” and “response to biotic stimulus”. Among the other findings, they are quite interesting as well. For example, “proteolysis” is one of biological functions being mentioned in the literature as one possible therapeutic target for acute myelogenous leukemia (Horwitz et al., 2003). It is also related to quite a few significant genes mentioned in Golub’s paper to dif-

Table 3.7: GO-level analysis results for the most significant GO terms found by DEA-PLS in Golub’s data.

<table>
<thead>
<tr>
<th>GO Term</th>
<th>O</th>
<th>A</th>
<th>Fisher’s Exact (Rank)</th>
<th>DEA-PLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 reactive oxygen species metabolism</td>
<td>7</td>
<td>45</td>
<td>4.24 × 10⁻⁵ (130)</td>
<td>3.00 × 10⁻⁵</td>
</tr>
<tr>
<td>2 response to oxidative stress</td>
<td>6</td>
<td>30</td>
<td>1.91 × 10⁻² (98)</td>
<td>1.45 × 10⁻²</td>
</tr>
<tr>
<td>3 nucleotide metabolism</td>
<td>17</td>
<td>113</td>
<td>3.13 × 10⁻³ (18)</td>
<td>2.94 × 10⁻¹8</td>
</tr>
<tr>
<td>4 defense response</td>
<td>80</td>
<td>730</td>
<td>7.77 × 10⁻³ (3)</td>
<td>7.63 × 10⁻¹8</td>
</tr>
<tr>
<td>5 immune response</td>
<td>69</td>
<td>664</td>
<td>1.18 × 10⁻³ (11)</td>
<td>5.63 × 10⁻¹8</td>
</tr>
<tr>
<td>6 regulation of cell adhesion</td>
<td>5</td>
<td>32</td>
<td>7.94 × 10⁻² (247)</td>
<td>5.63 × 10⁻¹8</td>
</tr>
<tr>
<td>7 regulation of MAPK activity</td>
<td>5</td>
<td>41</td>
<td>1.76 × 10⁻¹ (414)</td>
<td>5.63 × 10⁻¹8</td>
</tr>
<tr>
<td>8 cellular morphogenesis</td>
<td>15</td>
<td>181</td>
<td>3.40 × 10⁻¹ (645)</td>
<td>6.04 × 10⁻¹8</td>
</tr>
<tr>
<td>9 coenzyme metabolism</td>
<td>12</td>
<td>69</td>
<td>3.75 × 10⁻¹ (27)</td>
<td>6.04 × 10⁻¹8</td>
</tr>
<tr>
<td>10 response to biotic stimulus</td>
<td>84</td>
<td>751</td>
<td>2.30 × 10⁻⁵ (1)</td>
<td>6.04 × 10⁻¹8</td>
</tr>
<tr>
<td>11 proteolysis</td>
<td>9</td>
<td>37</td>
<td>2.35 × 10⁻⁵ (108)</td>
<td>7.52 × 10⁻¹8</td>
</tr>
</tbody>
</table>

A – the number of probe sets annotated for each GO term.
O – the number of significant probe sets annotated for each GO term.

Table 3.8: The most significant probe sets in Golub’s data.

<table>
<thead>
<tr>
<th>Probe ID</th>
<th>Gene Symbol</th>
<th>Raw p-values</th>
</tr>
</thead>
<tbody>
<tr>
<td>X05745_at</td>
<td>Zyxin</td>
<td>1.33 × 10⁻¹⁶</td>
</tr>
<tr>
<td>M23197_at</td>
<td>CD33</td>
<td>4.51 × 10⁻¹⁰</td>
</tr>
<tr>
<td>M84526_at</td>
<td>Adipsin</td>
<td>1.09 × 10⁻⁵</td>
</tr>
<tr>
<td>M16038_at</td>
<td>LYN</td>
<td>2.01 × 10⁻⁹</td>
</tr>
<tr>
<td>M22960_a_at</td>
<td>galactosialidosis</td>
<td>3.33 × 10⁻⁵</td>
</tr>
<tr>
<td>M63138_at</td>
<td>Cathepsin D</td>
<td>3.54 × 10⁻⁵</td>
</tr>
<tr>
<td>M35150_at</td>
<td>fumarylacetocacetate</td>
<td>5.24 × 10⁻⁴</td>
</tr>
<tr>
<td>M62762_at</td>
<td>ATPase</td>
<td>7.66 × 10⁻⁶</td>
</tr>
<tr>
<td>U90136_at</td>
<td>LTC4 synthase</td>
<td>1.25 × 10⁻⁷</td>
</tr>
</tbody>
</table>
ferentiate ALL and AML, such as “adipsin”, “galactosialidosis”, “cathepsin D” (Golub et al., 1999). In Table 3.8, the most significant probe sets are presented. The previously mentioned genes “adipsin”, “galactosialidosis”, “cathepsin D” are ranked fourth, sixth and seventh most significant. It is encouraging to see DEA-PLS recognize “proteolysis” as an important GO term even though Fisher’s exact test does not.

For Golub’s data, the findings are fairly consistent between DEA-PLS and Fisher’s exact test. Nonetheless, DEA-PLS finds not only GO terms with stronger signals (smaller p-values), but also some important GO terms Fisher’s exact test neglects. Some of these GO terms are mentioned in the literature and are used to identify different subtypes of leukemia. These results show us the effectiveness of DEA-PLS.

3.4 Conclusion

We focused our work on functional analysis instead of individual gene-level analysis. By using the gene ontology, DEA is able to increase interpretability of the analysis results without loss of sensitivity. As we presented in our analysis on Chiaretti’s data, the results can be displayed as a hierarchical structure. Hence, we are able to find clusters of nested GO terms which can be used to
differentiate leukemia subtypes.

Additionally, we addressed some of the problems of current pathway analysis methods. We propose to combine information and create a summary statistic for each GO term from the individual gene expressions, and then test for significance using the newly created variables. We considered two such summaries, mean and PLS score. As demonstrated in our simulation and analysis on two publicly available data sets, the new test has greater power to detect signals at the GO level than do methods such as Fisher’s exact test or GSEA. These methods lose power because they rely only on the rank of significance for individual genes. Furthermore, in our analysis on Chiaretti’s data set, we were able to elaborate that DEA-PLS maintains an appropriate order of importance for identified GO terms according to their actual gene expressions. Instead of testing for over-representation of each GO term only by the number of significant genes as Fisher’s exact test does, DEA-PLS considers the extent of importance of each gene related to the GO term by finding a linear combination of gene expressions using PLS. DEA-PLS has another advantage that it does not require gene-level analysis, and thus avoids dependence on the effectiveness of gene-level analysis.

DEA-PLS also can be considered as a type of dimension reduction. While GSEA and Fisher’s exact test provide summaries for each GO term, they do
so by giving a single number that represents all samples. DEA, on the other hand, provides summaries for each GO term and each sample, thus providing a more specific summarization. By creating summary measurements for each GO term, the new data usually have fewer variables than the original gene level data. These new variables represent meaningful biological functions instead of many individual genes. It could be advantageous to build a predictive model using these new variables because the predictive models would better reflect biological function and therefore be more interpretable.

In our analysis, we simply use a t-test to find significant GO terms after we combine the genes using PLS. It is suggested in the literature that the t distribution is not a valid null distribution for testing gene expression data. Our on-going research also shows that even when the normality assumption holds for the gene expression data, with a binary response the first score from PLS procedure is not normally distributed. This is partially the reason that we have excessive number of important GO terms with a biased null distribution. In the paper, we suggest to only focus on the most significant GO terms. We are currently working on adjusting the test according to the distributional information of the first score from PLS. An alternative approach is to use nonparametric testing instead of the t-test. Testing procedures such as the Significance Analysis of Microarray method (SAM) by Tusher et al. (2001),
empirical Bayes (EB) method of Efron et al. (2001) and the mixture model method (MMM) of Pan et al. (2001) are possibilities. These methods can potentially improve the performance of the testing part of our procedure. We are currently working on finding the right testing procedure to maximize the effectiveness of DEA methods.

One key aspect of DEA methods is that they are adaptive to many other analysis techniques. As we mentioned, after combining the variables by GO terms using PLS, we basically have a new set of data with fewer but more meaningful variables. We can either build a predictive model with GO terms, or improve our testing procedure with many existing methods for microarray data. We can also adapt DEA to other techniques, such as one proposed by Alexa et al. (2006), where they suggest improved scoring of GO terms by decorrelating GO graph structures. With much room to expand and improve, DEA is a promising new method for providing accurate and interpretable analysis of microarray data.
Chapter 4

Distribution of the First PLS Score for Binary Classification

4.1 Introduction

A general expression of the linear model is \( y = X\beta + \varepsilon \). Principal components analysis tries to find linear combinations of \( x_i \) that are orthogonal and maximizes \( c^T X^T X c \), while partial least squares seeks maximization of \( w^T X^T y c \).

For our case, the response \( y \) for an individual is a binary indicator of treatment or disease status. To keep things simple, we assume an equal number of cases and controls so that \( y = \left[ \begin{array}{c} 0_n^T \\ 1_n^T \end{array} \right]^T \), where \( 0_n \) is the \( n \times 1 \) vector of
zeroes and \( \textbf{1}_n \) is the \( n \times 1 \) vector of ones. A set of \( m \) genes provides covariate information. Let \( x_{kij} \) represent the expression level for gene \( j \) corresponding to individual \( i \) within treatment group \( y = k \), where \( k = 0, 1, i = 1, 2, \ldots, n \) and \( j = 1, \ldots, m \). Consequently, the design matrix for the linear model is

\[
X = \begin{bmatrix} X_0 \\ X_1 \end{bmatrix}, \quad X_k = \{x_{kij}\}_{i=1,2,\ldots,n,j=1,\ldots,m}, \text{ for } k = 0, 1. \quad (4.1)
\]

There are a total of \( 2n \) individuals or samples and \( m \) genes. Our goal is to test for differential expression of this set of genes according to treatment. Assuming \( x_{ki} = (x_{k1i}, \ldots, x_{kim})^T \) has mean \( \mu_k \) for \( k = 0, 1 \) and \( i = 1, \ldots, n \), we wish to test

\[
H_0 : \mu_0 = \mu_1 = 0, \quad \text{versus} \quad H_a : \mu_0 \neq \mu_1. \quad (4.2)
\]

Rather than focusing on the individual genes, we instead summarize the \( m \) gene expressions for each individual to get a single measure. Hypothesis testing is then conducted at this reduced dimension.

Dimension reduction by partial least squares (PLS) seeks a linear combination \( X\textbf{w}_1 \) such that the covariance with \( y \) is maximized. More specifically, PLS seeks \( \textbf{w}_1 \) to maximize \((\textbf{w}_1^T X^T y)^2\) such that \( ||\textbf{w}_1|| = 1 \); \( X\textbf{w}_1 \) is called the first PLS X-score or first PLS X-component and \( \textbf{w}_1 \) is its associated weight vector (de Jong, 1993). PLS decomposition typically continues by projection onto the
space of $Xw_1$ then finding $w_q$ to maximize covariance between the residual $X$ and $y$ spaces for $q = 2, 3, \cdots$. We are interested only in the first PLS $X$-score $Xw_1$. It is known that $\max_w (w^T X^T y)^2 = \max_w (w^T X^T y y^T X w)$ occurs when $w$ is the first left singular vector of $X^T y$ (Helland, 1988). Consequently, $w_1 = \frac{x^T y}{||X^T y||}$ and the first PLS $X$-score is

$$S = \frac{XX^T y}{||X^T y||}.$$ 

The $2n \times 1$ vector $S$ summarizes information provided by the $m$ gene expressions in a way that maximizes variability between treatment groups. In our proposed DEA-PLS, we perform a two sample t-test to compare $S$ with respect to $y = 0$ and $y = 1$. Our t-tests are based on

$$t = \frac{(\text{mean of } S \text{ for } y = 0) - (\text{mean of } S \text{ for } y = 1)}{\sqrt{\left(\frac{\text{variance of } S \text{ for } y = 0}{n}\right) + \left(\frac{\text{variance of } S \text{ for } y = 1}{n}\right)}}$$

$$= \frac{\overline{S}_0 - \overline{S}_1}{\sqrt{\frac{s^2_0}{n} + \frac{s^2_1}{n}}}$$

(4.3)

with two-sided rejection region $|t| > c$. We contend that Equation (4.3) is a test of the hypothesis (4.2) and determine the p-values as

$$\Pr_{\mu_0 = \mu_1}(|t| > |t_{\text{obs}}|).$$
For this test to be valid, we need the assumption that $S$ is from a normal distribution. While we have assumed $X$ follows a normal distribution, $S = \frac{XX^T y}{\|X^Ty\|}$ is clearly not from a normal distribution. In the reminder of this chapter, we demonstrate that the empirical null distribution of our test statistic $t$ is very different from a Student’s t distribution. We also improve the testing procedure by finding the true null distribution of $t$. The exact null distribution is obtained for small $m$ and we study different ways of approximating the null distribution for large $m$.

4.2 Theoretical Derivation of the Null Distribution

We next derive the null distribution for $t$ assuming

$$x_{ki} = (x_{ki1}, \cdots, x_{kim})^T \sim N(\mu_k, I)$$

(4.4)

for $k = 0, 1$ and $i = 1, \cdots, n$. To make the derivation easier at first, we assume genes are independent to each other and have been scaled to variance of one.
We then have

\[
\bar{S}_1 = \frac{1}{n} \left[ \begin{array}{cc} 0^T_n & 1^T_n \end{array} \right] X w_1 = \frac{1}{n} \left[ \begin{array}{cc} 0^T_n & 1^T_n \end{array} \right] \frac{X X^T y}{||X^T y||} = \frac{1}{n} \frac{y^T X X^T y}{||X^T y||} = \frac{||X^T y||}{n},
\]

and

\[
\bar{S}_0 = \frac{1}{n} \left[ \begin{array}{cc} 1^T_n & 0^T_n \end{array} \right] X w_1 = \frac{1}{n} \left( 1^T_{2n} - y^T \right) \frac{X X^T y}{||X^T y||} = \frac{1}{n} \left[ \frac{1^T X X^T y}{||X^T y||} - ||X^T y|| \right].
\]

In fact,

\[
\bar{S}_0 - \bar{S}_1 = \frac{1}{n} \left[ \begin{array}{cc} 1^T_n & -1^T_n \end{array} \right] X w_1 = \frac{1}{n} \left[ \begin{array}{cc} 1^T_n & -1^T_n \end{array} \right] X X^T y \frac{1}{||X^T y||} = \frac{1}{n} \left[ \begin{array}{c} \left( 1^T_n - 1^T_n \right) X \right] \left( X^T y \right) \frac{1}{||X^T y||}
\]

\[
= \left[ (n \overline{X}_{0,1} - n \overline{X}_{1,1}) , \ldots , (n \overline{X}_{0,m} - n \overline{X}_{1,m}) \right] \left[ \begin{array}{c} n \overline{X}_{1,1} \\vdots \\vdots \\vdots \n \overline{X}_{1,m} \end{array} \right] \frac{1}{n ||X^T y||} \tag{4.5}
\]

where \( \overline{X}_{k,j} = \frac{1}{n} (x_{k1j} + \cdots + x_{knj}) \) is the mean expression level for gene \( j \) in treatment group \( y = k \).
Note that

\[ n||X^Ty||^2 (\overline{S}_0 - \overline{S}_1) = n^2 \sum_{j=1}^{m} (\overline{X}_{0,j} - \overline{X}_{1,j}) \overline{X}_{1,j} \]

\[ = nW^m, \]

where

\[ W^m \equiv n \sum_{j=1}^{m} (\overline{X}_{0,j} - \overline{X}_{1,j}) \overline{X}_{1,j} = n \sum_{j=1}^{m} \overline{X}_{0,j} \overline{X}_{1,j} - n \sum_{j=1}^{m} \overline{X}_{1,j}^2. \]  \hspace{1cm} (4.6)

For \( H_0: \mu_0 = \mu_1 = 0 \), and distributional assumption (4.4), we have

\[ \overline{X}_{k,j} \sim N(0, \frac{1}{n}), k = 0, 1, j = 1, \ldots, m \]

\[ \Rightarrow \left\{ \begin{array}{l}
\frac{1}{n}(\overline{X}_{1,1} \overline{X}_{1,1}), \ldots, n(\overline{X}_{1,m} \overline{X}_{1,m}) \sim \chi^2_1 \\
\frac{1}{n}(\overline{X}_{0,1} \overline{X}_{1,1}), \ldots, n(\overline{X}_{0,m} \overline{X}_{1,m}) \sim f_p
\end{array} \right. \]  \hspace{1cm} (4.7)

where \( f_p \) is the product of two independent \( N(0,1) \).

Therefore, Equation (4.6) and (4.7) \( \Rightarrow \)

\[ W^m = \frac{d}{i=1} f_p - \sum_{i=1}^{m} \chi^2_1 = \sum_{i=1}^{m} f_p - \chi^2_m. \]  \hspace{1cm} (4.8)

What is \( f_p \)? Let us say \( X_1 \sim N(\mu_1, \sigma_1^2), X_2 \sim N(\mu_2, \sigma_2^2) \), \( X_1 \) and \( X_2 \) are
independent, and define $Y = X_1X_2$. What is the distribution of $f(y)$?

$$f(x_1, x_2) = \frac{1}{2\pi \sigma_1 \sigma_2} \exp \left( -\frac{1}{2} \left( \frac{x_1 - \mu_1}{\sigma_1} \right)^2 - \frac{1}{2} \left( \frac{x_2 - \mu_2}{\sigma_2} \right)^2 \right), \text{ for all } x_1, x_2 \in R.$$  

We know $Y|X_2 \sim N(X_2 \mu_1, X_2^2 \sigma_1^2)$, and $X_2 \sim N(\mu_2, \sigma_2^2)$. Therefore,

$$f(y, x_2) = f(y|x_2) f(x_2) = \frac{1}{2\pi |x_2| \sigma_1 \sigma_2} \exp \left( -\frac{1}{2\sigma_1^2} \left( \frac{y}{x_2} - \mu_1 \right)^2 - \frac{1}{2\sigma_2^2} (x_2 - \mu_2)^2 \right).$$

We have

$$f(y) = \int_{-\infty}^{\infty} f(y, x_2) dx_2$$

$$= \int_{-\infty}^{\infty} f(y|x_2) f(x_2) dx_2$$

$$= \int_{-\infty}^{\infty} \frac{1}{2\pi |x_2| \sigma_1 \sigma_2} \exp \left( -\frac{1}{2\sigma_1^2} \left( \frac{y}{x_2} - \mu_1 \right)^2 - \frac{1}{2\sigma_2^2} (x_2 - \mu_2)^2 \right) dx_2.$$  

This cannot be easily integrated.

Let us go back to Equation (4.8) and look at the case when $m = 1$. Rather than use notation $W^1$ for $W^m$ when $m = 1$, we will instead use $W_1$, i.e,

$$W_1 \overset{d}{=} f_p - \chi_1^2 = X_1 X_2 - X_1^2,$$  

(4.9)

where $X_1 = \sqrt{nX_{0.1}}$ and $X_2 = \sqrt{nX_{1.1}}$ are independent and from $N(0, 1)$. We redefine the notation of $X_1$, $X_2$ here just for transformation.
Let $U = X_1 X_2$ and $V = X_1^2$. Since the transformation is not one-one, we define $A_1 = \{(x_1, x_2) : x_1 > 0\}$, $A_2 = \{(x_1, x_2) : x_1 < 0\}$, and $A_3 = \{(x_1, x_2) : x_1 = 0\}$. Let $B = \{(u, v) : v > 0\}$. Therefore as $B \to A_1$,

\[
\begin{cases}
X_1 = \sqrt{V} \\
X_2 = \frac{u}{\sqrt{V}}
\end{cases}
\]

\[
J_1 = \begin{vmatrix}
\frac{\partial X_1}{\partial U} & \frac{\partial X_1}{\partial V} \\
\frac{\partial X_2}{\partial U} & \frac{\partial X_2}{\partial V}
\end{vmatrix} = \begin{vmatrix} 0 & \frac{1}{2} V^{-\frac{1}{2}} \\ \frac{1}{\sqrt{V}} & -\frac{u}{2} V^{-\frac{3}{2}} \end{vmatrix} = -\frac{1}{2V}.
\]

As $B \to A_2$,

\[
\begin{cases}
X_1 = -\sqrt{V} \\
X_2 = -\frac{u}{\sqrt{V}}
\end{cases}
\]

\[
J_2 = \begin{vmatrix}
\frac{\partial X_1}{\partial U} & \frac{\partial X_1}{\partial V} \\
\frac{\partial X_2}{\partial U} & \frac{\partial X_2}{\partial V}
\end{vmatrix} = \begin{vmatrix} 0 & -\frac{1}{2} V^{-\frac{1}{2}} \\ -\frac{1}{\sqrt{V}} & -\frac{u}{2} V^{-\frac{3}{2}} \end{vmatrix} = -\frac{1}{2V}.
\]

Since

\[
f(x_1, x_2) = \frac{1}{2\pi \sigma_1 \sigma_2} \exp \left( -\frac{1}{2} \left( \frac{x_1 - \mu_1}{\sigma_1} \right)^2 - \frac{1}{2} \left( \frac{x_2 - \mu_2}{\sigma_2} \right)^2 \right)
\]
therefore,

\[
f(u, v) = \frac{1}{2\pi \sigma_1 \sigma_2} \exp \left( -\frac{1}{2} \left( \frac{\sqrt{v} - \mu_1}{\sigma_1} \right)^2 - \frac{1}{2} \left( \frac{u}{\sqrt{v}} \right)^2 \right) \left| \frac{1}{2v} \right| + \frac{1}{2\pi \sigma_1 \sigma_2} \exp \left( -\frac{1}{2} \left( \frac{-\sqrt{v} - \mu_1}{\sigma_1} \right)^2 - \frac{1}{2} \left( \frac{-u}{\sqrt{v}} \right)^2 \right) \left| \frac{1}{2v} \right|
\]

\[
= \frac{1}{2\pi} \exp \left( -\frac{1}{2} \left( \sqrt{v} \right)^2 - \frac{1}{2} \left( \frac{u}{\sqrt{v}} \right)^2 \right) \frac{1}{2v} + \frac{1}{2\pi} \exp \left( -\frac{1}{2} \left( \sqrt{v} \right)^2 - \frac{1}{2} \left( \frac{u}{\sqrt{v}} \right)^2 \right) \frac{1}{2v}
\]

as \( \mu_1 = \mu_2 = 0, \sigma_1 = \sigma_2 = 1 \) and \(-\infty < u < \infty, v \geq 0\)

\[
= \frac{1}{4\pi v} \exp \left( -\frac{1}{2} \left( v + \frac{u^2}{v} \right) \right) + \frac{1}{4\pi v} \exp \left( -\frac{1}{2} \left( v + \frac{u^2}{v} \right) \right)
\]

\[
= \frac{1}{2\pi} \exp \left( -\frac{v^2 + u^2}{2v} \right) \quad (4.10)
\]

Now that we have the joint density of \( U \) and \( V \), we can find \( W_1 = U - V \) that we are truly interested in.

Let

\[
\begin{cases}
W_1 = U - V \
Z = V
\end{cases}
\Rightarrow \begin{cases}
U = W_1 + Z \\
V = Z
\end{cases}
\]

\[
J = \begin{vmatrix}
\frac{\partial U}{\partial W_1} & \frac{\partial U}{\partial Z} \\
\frac{\partial V}{\partial W_1} & \frac{\partial V}{\partial Z}
\end{vmatrix} = \begin{vmatrix}
1 & 1 \\
0 & 1
\end{vmatrix} = 1.
\]

103
Therefore from Equation (4.10), we get

\[
f(w_1, z) = \frac{1}{2z\pi} \exp \left( -\frac{(w_1 + z)^2 + z^2}{2z} \right), -\infty < w < \infty, 0 < z
\]

\[
= \frac{1}{2z\pi} \exp \left( -\frac{w_1^2 + 2w_1z + 2z^2}{2z} \right).
\]

Therefore

\[
f(w_1) = \int_0^\infty \frac{1}{2z\pi} \exp \left( -\frac{w_1^2 + 2w_1z + 2z^2}{2z} \right) dz
\]

\[
= \frac{\exp(-w_1)\text{BesselK}(0, \sqrt{2\sqrt{w_1^2}})}{\pi}, \tag{4.11}
\]

where \(\text{BesselK}(n, x)\) is a modified bessel function of the second kind. As a special case of \(n = 0\),

\[
\text{BesselK}(0, x) = \int_0^\infty \cos(x \sinh t) dt = \int_0^\infty \frac{\cos(xt)}{\sqrt{t^2 + 1}} dt. \tag{4.12}
\]

The density is plotted in Figure 4.1. Using maple, we are able to find the 5th percentile \(a = -4.45\) such that \(\text{Pr}(W_1 < a) = \alpha = 0.0501\). Notice that the density function is skewed to the left, and hence is clearly not the Student’s t density.

We can find the distribution of \(W_1 \sim f(w_1)\) from Equation (4.11). Now, we
Figure 4.1: Theoretical density plot of $W_1$

rewrite Equation (4.8) as

$$W^m \sim \sum_{i=1}^{m} f_p - \sum_{i=1}^{m} \chi^2_1 = \sum_{i=1}^{m} (f_p - \chi^2_1) = \sum_{i=1}^{m} (W_i), \quad (4.13)$$

where $W_i$ are independently distributed from density $f(w_1)$ of Equation (4.11).
Let us try to find the density of $W^2$. We have

$$f(w_1) = \frac{\exp(-w_1) \text{BesselK}(0, \sqrt{2} \sqrt{w_1^2})}{\pi},$$

and

$$f(w_2) = \frac{\exp(-w_2) \text{BesselK}(0, \sqrt{2} \sqrt{w_2^2})}{\pi}.$$

Hence,

$$f(w_1, w_2) = \frac{\exp(-w_1) \text{BesselK}(0, \sqrt{2} \sqrt{w_1^2})}{\pi} \frac{\exp(-w_2) \text{BesselK}(0, \sqrt{2} \sqrt{w_2^2})}{\pi}.$$

Let $U = W_1 + W_2$ and $V = W_1 \Rightarrow W_1 = V$ and $W_2 = U - V \Rightarrow J = 1$.

Hence,

$$f(u, v) = \frac{\exp(-v) \text{BesselK}(0, \sqrt{2} \sqrt{v^2}) \exp(-(u - v)) \text{BesselK}(0, \sqrt{2} \sqrt{(u - v)^2})}{\pi} \frac{\pi}{\pi} = \frac{\exp(-u) \text{BesselK}(0, \sqrt{2} \sqrt{u^2}) \text{BesselK}(0, \sqrt{2} \sqrt{(u - v)^2})}{\pi^2}.$$

Therefore,

$$f(u) = \int_{-\infty}^{\infty} \frac{\exp(-u) \text{BesselK}(0, \sqrt{2} \sqrt{v^2}) \text{BesselK}(0, \sqrt{2} \sqrt{(u - v)^2})}{\pi^2} dv.$$

Using maple, we can approximate the density function and plot in Figure 4.2. It takes 14 hours to approximate and plot this figure, but the final result is not
complete. For comparison purpose, smoothed density functions obtained from the simulated data under the null are given in Figure 4.3 with different size of gene sets. The details of simulating the empirical distribution are explained in Section 4.4. From the figure, we find the skewness and variance of the null distributions increases as \( m \) increases. It is clear that we cannot simply use a single t-distribution as the null distribution to test for all gene sets. The null distribution needs to be derived according to the size of the gene sets.

Figure 4.2: The theoretical Density plot of \( W^2 \)
4.3  Approximations of the Null Distribution

4.3.1  Normal Approximation by the Central Limit Theorem

It is impractical to exactly determine the distribution of $W^m$ for $m > 2$ by transformation. We revisit Equation (4.13) and realize that $W_1, \ldots, W_m$ are
iid from density \( f(w_1) \) of Equation (4.11). We can try to approximate \( W^m \) using the central limit theorem. By the CLT, as \( m \to \infty \), we approximately have

\[
\frac{\sqrt{m} \left( \frac{1}{m} \sum_{i=1}^{m} W_i - \mu \right)}{\sigma} \xrightarrow{d} N(0, 1) \Rightarrow \frac{1}{m} \sum_{i=1}^{m} W_i \approx N(\mu, \frac{\sigma^2}{m})
\]

\[
\Rightarrow W^m = \sum_{i=1}^{m} W_i \approx N(m\mu, m\sigma^2),
\]

where the true mean and variance are \( \mu = E(W_1) = -1 \) and \( \sigma^2 = E(W_1^2) - (E(W_1))^2 = 3 \), computed by maple using Equation (4.11).

To see how well it is to approximate \( W^m \) using normal distribution, we simulate the distribution of \( W^m \). We first generate \( X_{1i}, X_{2i} \) \( iid \sim N(0, 1) \), and then form \( W_i = X_{1i}X_{2i} - X_{1i}^2 \) for \( i = 1, \ldots, m \). Finally we compute \( W^m = \sum_{i=1}^{m} W_i \). This procedure is repeated for 20000 times to generate the empirical null distributions. The 5th sample percentile from the 20000 random samples are shown with the normal approximation in Table 4.1. It is expected that the normal approximation does not do well when \( m \) is small, but even for \( m \) as large as 60, the normal approximation is still not performing very well at the tail area. It is more clear for comparison if we standardize the distribution of
Let

\[ Y_m = \frac{W^m - m\mu}{\sigma \sqrt{m}} \]

and \( Y_m \) can be approximated as standard normal as \( m \) grows larger. Figure 4.4 displays the 5th percentile of empirical distribution of \( Y_m \) and its standard normal approximation. It shows us clearly how the standard normal approximation of \( Y_m \) improves as \( m \) grows larger, but it is still not adequate even for moderate \( m \).

One other possibility is to approximate the distribution using the skew-normal (Azzalini and Capitanio, 1999). See Figures 4.5–4.8. The skew-normal curves are fit with simulated data for \( f(w_m) \) using maximum likelihood methods. We use function “sn.mle” in R package “sn” to obtain the skew-normal fits. The approximation is fairly good at the tail area. For \( W_1 \), the 5th percentile obtained from the fitted skew-normal is \(-4.123\) compared to the exact 5th percentile of \(-4.45\) obtained from Equation (4.11). Similarly, the 5th percentile for the fitted skew-normal are \(-6.549, -8.631, -10.579\) as \( m = 2, m = 3, m = 4 \) compared to percentiles estimated from simulated empirical distributions of \(-6.75, -8.73, -10.83\). The fitted curves demonstrate that the skew-normal can approximate \( f(w^m) \) much better than the first-order approximation based on the CLT. From previous results, we know we can find the cumulant generating function for \( f(w_1) \) and hence compute the cumulants of \( W_1 \). In later
Table 4.1: Comparing 5th percentile of simulated $W^m$ and $N(-1m, 3m)$.

<table>
<thead>
<tr>
<th>$m$</th>
<th>mean($W^m$)</th>
<th>var($W^m$)</th>
<th>5th percentile of $W^m$</th>
<th>5th percentile of $N(-1m, 3m)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1.03</td>
<td>3.10</td>
<td>-4.59173</td>
<td>-3.84897</td>
</tr>
<tr>
<td>2</td>
<td>-2.00</td>
<td>5.88</td>
<td>-6.75204</td>
<td>-6.02905</td>
</tr>
<tr>
<td>3</td>
<td>-2.98</td>
<td>8.63</td>
<td>-8.73089</td>
<td>-7.93456</td>
</tr>
<tr>
<td>4</td>
<td>-4.03</td>
<td>12.42</td>
<td>-10.83475</td>
<td>-9.69794</td>
</tr>
<tr>
<td>5</td>
<td>-5.06</td>
<td>14.87</td>
<td>-12.35761</td>
<td>-11.37049</td>
</tr>
<tr>
<td>6</td>
<td>-6.10</td>
<td>18.57</td>
<td>-14.14867</td>
<td>-12.97852</td>
</tr>
<tr>
<td>7</td>
<td>-7.06</td>
<td>20.78</td>
<td>-15.49733</td>
<td>-14.53767</td>
</tr>
<tr>
<td>8</td>
<td>-8.01</td>
<td>23.86</td>
<td>-17.09763</td>
<td>-16.05810</td>
</tr>
<tr>
<td>9</td>
<td>-9.02</td>
<td>27.32</td>
<td>-18.59001</td>
<td>-17.54691</td>
</tr>
<tr>
<td>10</td>
<td>-10.05</td>
<td>30.49</td>
<td>-20.21900</td>
<td>-19.00923</td>
</tr>
<tr>
<td>11</td>
<td>-10.97</td>
<td>32.73</td>
<td>-21.44372</td>
<td>-20.44897</td>
</tr>
<tr>
<td>12</td>
<td>-11.89</td>
<td>35.62</td>
<td>-22.95797</td>
<td>-21.86912</td>
</tr>
<tr>
<td>14</td>
<td>-14.03</td>
<td>42.83</td>
<td>-25.91520</td>
<td>-24.65987</td>
</tr>
<tr>
<td>15</td>
<td>-15.08</td>
<td>45.88</td>
<td>-27.48051</td>
<td>-26.03401</td>
</tr>
<tr>
<td>16</td>
<td>-16.08</td>
<td>49.32</td>
<td>-28.59748</td>
<td>-27.39588</td>
</tr>
<tr>
<td>17</td>
<td>-16.97</td>
<td>52.09</td>
<td>-30.09330</td>
<td>-28.74660</td>
</tr>
<tr>
<td>18</td>
<td>-18.01</td>
<td>55.11</td>
<td>-31.45824</td>
<td>-30.08716</td>
</tr>
<tr>
<td>19</td>
<td>-18.99</td>
<td>57.41</td>
<td>-32.79430</td>
<td>-31.41837</td>
</tr>
<tr>
<td>20</td>
<td>-20.10</td>
<td>61.40</td>
<td>-34.27475</td>
<td>-32.74098</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>48</td>
<td>-47.95</td>
<td>143.82</td>
<td>-68.988330</td>
<td>-67.73824</td>
</tr>
<tr>
<td>49</td>
<td>-49.02</td>
<td>148.01</td>
<td>-70.09339</td>
<td>-68.94279</td>
</tr>
<tr>
<td>50</td>
<td>-49.94</td>
<td>149.22</td>
<td>-71.16966</td>
<td>-70.14526</td>
</tr>
<tr>
<td>51</td>
<td>-50.98</td>
<td>154.04</td>
<td>-72.56607</td>
<td>-71.34572</td>
</tr>
<tr>
<td>52</td>
<td>-51.96</td>
<td>154.50</td>
<td>-73.68831</td>
<td>-72.54422</td>
</tr>
<tr>
<td>53</td>
<td>-53.11</td>
<td>158.41</td>
<td>-74.77093</td>
<td>-73.74082</td>
</tr>
<tr>
<td>54</td>
<td>-54.04</td>
<td>163.57</td>
<td>-76.30803</td>
<td>-74.93557</td>
</tr>
<tr>
<td>55</td>
<td>-55.02</td>
<td>164.93</td>
<td>-77.28317</td>
<td>-76.12853</td>
</tr>
<tr>
<td>56</td>
<td>-55.89</td>
<td>168.13</td>
<td>-78.32012</td>
<td>-77.31974</td>
</tr>
<tr>
<td>57</td>
<td>-56.99</td>
<td>171.23</td>
<td>-79.86858</td>
<td>-78.50925</td>
</tr>
<tr>
<td>58</td>
<td>-57.97</td>
<td>172.12</td>
<td>-80.81539</td>
<td>-79.69711</td>
</tr>
<tr>
<td>59</td>
<td>-58.90</td>
<td>179.35</td>
<td>-82.11782</td>
<td>-80.88335</td>
</tr>
<tr>
<td>60</td>
<td>-60.06</td>
<td>177.35</td>
<td>-83.18967</td>
<td>-82.06803</td>
</tr>
</tbody>
</table>
Figure 4.4: 5th percentile of empirical distribution of $Y_m = \frac{W^m - m\mu}{\sigma\sqrt{m}}$ displayed as dots and its standard normal approximation displayed as a line.

sections, we calculate the cumulants for $W^m$, then approximate its distribution by matching moments.
4.3.2 Approximation by Edgeworth Series Expansion

We now try to approximate the null distribution using an Edgeworth series expansion. Edgeworth expansions are higher order approximations to the distribution of a sum of independently identically distributed random variables obtained by Taylor expansion of the cumulant generating function. Instead of approximating the distribution of $W^m$, we work with the standardized version
Figure 4.6: Approximation of $f(w^2)$ by skew-normal distribution: the histogram is the empirical distribution of $W^2$ and the curve is the skew-normal fit. The exact 5th percentile is -6.75, compared to -6.549 from the skew normal fit.

of $W^m$ from now on because this facilitates comparisons for various $m$:

$$Y_m = \frac{W^m - m\mu}{\sigma \sqrt{m}}.$$
Figure 4.7: Approximation of $f(w^3)$ by skew-normal distribution: the histogram is the empirical distribution of $W^3$ and the curve is the skew-normal fit. The exact 5th percentile is -8.73, compared to -8.631 from the skew normal fit.

To recall our setup, we have $W_1, \cdots, W_m$ independent and identically distributed random variables, $E(W_1) = \mu = -1, Var(W_1) = \sigma^2 = 3$. The cu-
Figure 4.8: Approximation of $f(w^4)$ by skew-normal distribution: the histogram is the empirical distribution of $W^4$ and the curve is the skew-normal fit. The exact 5th percentile is -10.83, compared to -10.579 from the skew normal fit.

The characteristic generating function for $f(w)$ in Equation (4.11) can be written as

$$K(W_1; t) = \log(E(\exp(tW_1)))$$

$$= \log \int_{-\infty}^{\infty} \exp(tw)f(w)dw$$

$$= \log \int_{-\infty}^{\infty} \frac{\exp(\frac{16-w}{\sqrt{2}\sqrt{w^2}})\text{BesselK}(0, \sqrt{2}\sqrt{w^2})}{\pi} dw$$
The cumulants are the derivatives of $K(t)$ evaluated at $t = 0$:

$$\kappa_1 = K'(0), \kappa_2 = K''(0), \ldots, \kappa_r = K^{(r)}(0).$$

Evaluating the first several cumulants of $W_1$ using Maple, we get:

$$\kappa_1 = -1, \kappa_2 = 3, \kappa_3 = -14, \kappa_4 = 102, \ldots$$

The cumulant generating function of $W_1$ can also be written as

$$K(W_1; t) = \mu t + \frac{1}{2} \sigma^2 t^2 + \frac{1}{6} \kappa_3 t^3 + \frac{1}{24} \kappa_4 t^4 + \cdots$$

Furthermore, the cumulant generating function of $Y_m$ can be written as (Barndorff-Nielsen and Cox, 1989; Severini, 2000):

$$K(Y_m; t) = -\sqrt{m} \frac{\mu t}{\sigma} + m K\left(W_1; \frac{t}{\sigma \sqrt{m}}\right).$$

After substituting Equation (4.14) into Equation (4.15) and some arithmetic, we get the expansion:

$$K(Y_m; t) = \frac{1}{2} t^2 + \frac{\kappa_3 t^3}{6 \sqrt{m} \sigma^3} + \frac{\kappa_4 t^4}{24 m \sigma^4} + O\left(m^{-\frac{3}{2}}\right).$$
The moment generating function can be approximated by:

\[
M(Y_m; t) = \exp \left( \frac{t^2}{2} \right) \left( 1 + \frac{\kappa_3 t^3}{6\sqrt{m} \sigma^3} + \frac{\kappa_4 t^4}{24m \sigma^4} + \frac{\kappa_5^2 t^6}{72m \sigma^6} \right) + O \left( m^{-\frac{3}{2}} \right). \tag{4.17}
\]

The moment generating function can be inverted to obtain an expansion of the probability density function of \( Y_m \) (Barndorff-Nielsen and Cox, 1989; Severini, 2000):

\[
f(Y_m; x) = \phi(x) \left( 1 + \frac{\kappa_3 H_3(x)}{6\sqrt{m} \sigma^3} + \frac{\kappa_4 H_4(x)}{24m \sigma^4} + \frac{\kappa_5^2 H_6(x)}{72m \sigma^6} \right) + O \left( m^{-\frac{3}{2}} \right), \tag{4.18}
\]

where \( H_j(x) \) is the \( j \)th Hermite polynomial, defined by \( H_r(x)\phi(x) = (-1)^r \frac{d^r \phi(x)}{dx^r} \).

The first several such polynomials are \( H_2(x) = x^2 - 1 \), \( H_3(x) = x^3 - 3x \), \( H_4(x) = x^4 - 6x^2 + 3 \), \( H_5(x) = x^5 - 10x^3 + 15x \), \( H_6(x) = x^6 - 15x^4 + 45x^2 - 15 \)
and so on. We substitute the cumulants into the density function and simplify them to get:

\[
f(Y_m; y) = \phi(y) \{ 0.1008 \frac{1}{m} y^6 - 0.10401 \frac{1}{m} y^4 + 0.4491 \frac{1}{\sqrt{m}} y^3 + 1.7037 \frac{1}{m} y^2 + \\
1.3472 \frac{1}{\sqrt{m}} y + (1 - 0.0956 \frac{1}{m}) \} + O \left( m^{-\frac{3}{2}} \right)
\]

Using Maple, we can easily find the 1st percentile and 5th percentile cutoff values from the density function for different value of \( m \). The density function here is an approximation of the distribution of \( Y_m \) accurate to order \( O(m^{-\frac{3}{2}}) \).
To achieve more accurate approximations, more terms need to be kept in Equation (4.16).

4.3.3 Skew-t Approximation Using the First Four Moments

From previous simulation results, we realize that the skew-normal distribution can be a good approximation of the distribution of \( Y_m \). The skew-normal fit requires availability of an empirical distribution function, which can be obtained by simulation. However, we do not want to rely on fitting simulated data to acquire the approximating distribution. When applying Edgeworth series approximation, we developed a tool to compute the cumulants of \( Y_m \) with Equation (4.16), the cumulant generating function of \( Y_m \). Thus, the first several cumulants of \( Y_m \) can be computed as:

\[
K'(Y_m; t = 0) = t + \frac{\kappa_3 t^2}{2\sqrt{m}\sigma^3} + \frac{\kappa_4 t^3}{6m\sigma^4} + \cdots |_{t=0} = 0
\]

\[
K''(Y_m; t = 0) = 1 + \frac{\kappa_3 t}{\sqrt{m}\sigma^3} + \frac{\kappa_4 t^2}{2m\sigma^4} + \cdots |_{t=0} = 1
\]

\[
K'''(Y_m; t = 0) = \frac{\kappa_3}{\sqrt{m}\sigma^3} + \frac{\kappa_4 t}{m\sigma^4} + \cdots |_{t=0} = \frac{\kappa_3}{\sigma^3\sqrt{m}}
\]

\[
K^{(4)}(Y_m; t = 0) = \frac{\kappa_4}{m\sigma^4} + \cdots |_{t=0} = \frac{\kappa_4}{\sigma^4 m}
\]
Notice that when these derivatives of the cumulant generating function are evaluated at $t = 0$, the remaining terms not shown are in fact zero. The cumulants computed here are not an approximation but the actual cumulants of $Y_m$. With a parametric assumption, we can fit a good approximating distribution of $Y_m$ with the computed cumulants using the method of moments.

We assume that $Y_m$ follows a skew-t distribution (Azzalini and Capitanio, 2003) which can be derived as a mixture of skew-normals. A random variable $Z$ with the standard skew-normal distribution and shape parameter $\alpha$ has density

$$f(z) = 2\phi(z)\Phi(\alpha z), \text{ for } -\infty < z < \infty,$$

where $\phi(z)$ is the standard-normal density $\phi(z) = \frac{1}{\sqrt{2\pi}} \exp \left(-\frac{z^2}{2}\right)$, and $\Phi(z)$ is the standard-normal distribution function $\Phi(z) = \int_{-\infty}^{z} \phi(t) dt$. A random variable $X$ with the skew-normal distribution with location, shape and scale parameters, $\epsilon$, $\alpha$, and $\sigma$ respectively, has density given by

$$f(z) = 2\phi(z) \left(\frac{x - \epsilon}{\sigma}\right) \frac{1}{\sigma} \Phi\left(\alpha \frac{x - \epsilon}{\sigma}\right).$$

A random variable with the skew-t distribution arises as a transformation of a skew-normal. Suppose $Z \sim SN(\alpha)$ and $V \sim \Gamma\left(\frac{\nu}{2}, \frac{\nu}{2}\right)$, i.e. $\chi^2/\nu$, then the
random variable

\[ T = \epsilon + \sigma \frac{Z}{\sqrt{\nu}} \]

has the skew-t distribution. Let \( f_T(t; \nu) \) and \( F_T(t; \nu) \) denote the density and distribution function for the usual t distribution with \( \nu \) degrees of freedom, then the density of the skew-t random variable \( T \) is given by

\[
f(t) = \frac{2}{\alpha} f_T \left( \frac{t - \epsilon}{\sigma} \right) f_T \left( \alpha \left( \frac{t - \epsilon}{\sigma} \right) \sqrt{\frac{\nu + 1}{\nu + (t - \nu)^2}; \nu + 1} \right).
\]

Skew-t has four parameters—location \( \epsilon \), scale \( \sigma \), shape \( \alpha \) and degrees of freedom \( \nu \). When the degrees of freedom go to infinity, skew-t becomes a skew-normal distribution. In other words, the skew-normal is a special case of skew-t distribution.

We can compute the first four cumulants of the general skew-t distribution. The first two cumulants of the skew-t distribution with parameters \( \epsilon = 0 \) and \( \sigma = 1 \) are (Azzalini and Capitanio, 2003)

\[
\kappa_1^* = \frac{\alpha}{\sqrt{1 + \alpha^2}} \sqrt{\frac{\nu}{\pi}} \Gamma \left( \frac{\nu - 1}{2} \right) \Gamma \left( \frac{\nu}{2} \right)
\]

\[
\kappa_2^* = \frac{\nu}{\nu - 2} - (\kappa_1^*)^2.
\]

Hence, the first four cumulants of the general skew-t distribution with four parameters—location \( \epsilon \), scale \( \sigma \), shape \( \alpha \) and degrees of freedom \( \nu \) are computed
As:

\[
\begin{align*}
\kappa_1 &= \epsilon + \sigma \kappa_1^* \\
\kappa_2 &= \sigma^2 \left( \frac{\nu}{\nu - 2} - \frac{(\kappa_1^*)^2}{\sigma^2} \right) \\
\kappa_3 &= \kappa_1^* \sigma^3 \left( \frac{\nu}{\nu - 3} \left( 3 - \frac{\alpha^2}{1 + \alpha^2} \right) - 3 \frac{\nu}{\nu - 2} + 2(\kappa_1^*)^2 \right) \\
\kappa_4 &= \sigma^4 \left( \frac{3\nu^2}{(\nu - 2)(\nu - 4)} - 4(\kappa_1^*)^2 \nu \left( \frac{3 - \frac{\sigma^2}{1 + \sigma^2}}{\nu - 3} \right) + 6(\kappa_1^*)^2 \frac{\nu}{\nu - 3} - 3(\kappa_1^*)^4 - 3(\kappa_2^*)^2 \right)
\end{align*}
\]

We can match these four cumulants of the skew-t distribution with the computed cumulants of \(Y_m\) using methods of moments. We simply solve the system of equations to find the four parameters of the fitted skew-t distribution. We use the R function “st.cumulants.inversion” in “sn” package to do this moment matching procedure. From the results in Figure 4.5–4.8, it is expected that skew-t approximation should perform fairly well when \(m\) is larger than 2.

### 4.4 Simulation Studies

#### 4.4.1 Comparison of the Approximation Methods

Three of the methods mentioned above can be used to approximate the distribution of \(Y_m\), normal approximation using CLT, Edgeworth series approximation by inverting the cumulant generating function and skew-t approx-
imation by method of moments. To assess which of these methods have the best approximation, we can compare them with empirical null distributions obtained from simulation.

To generate the empirical null distributions of our test statistics, we first generate $m$ gene expressions from $N(0, 1)$ with sample size of 100. Each of these genes are generated from one normal distribution instead of a mixture of normal because under the null these genes are not differentially expressed. The $m$ gene expressions are independently generated. We compute the test statistic $Y_m$ for this gene set to get one sampled value for the empirical null distribution of our test statistic. We then repeat this process, simulating 20000 sets of $m$ genes and compute the test statistics for each gene sets, to obtain the empirical null distributions. Different $m$ is used for each empirical null distributions with respect to different size of gene sets. Since we care more about the tail area of the distribution for testing purpose, we compare the 5th and 1st percentile of the empirical distribution to each of the approximations under study. (See Appendix D for R codes that generates the empirical distribution)

The results for the 5th percentile are shown in Figure 4.9 and 1st percentile in Figure 4.10. In the figures, the circles represent the percentiles obtained from the simulated empirical distribution. The straight line, dashed line and dotted line represent the percentiles for normal, skew-t and Edgeworth approx-
First of all, we know the true distribution of $Y_m$ converges to normal when $m$ goes to infinity according to CLT. In the figure, the dots from (simulated distribution) are certainly approaching the straight line (from the normal distribution). However, the convergence is very slow, as even at $m = 60$ the 5th percentiles of simulated distribution are still far from those of normal. On the same note, the Edgeworth series approximation also performs much better than the normal approximation when $m$ is small. Since we only have four terms...
Figure 4.10: 1st percentile of $Y_m$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 5th percentiles of the empirical distribution of $Y_m$ based on 20000 simulated data points.

In our cumulant generating function, the approximation can be improved by adding more terms to the function. Of course, that can make the computation more complicated. The Edgeworth approximation by itself should converge to normal, which is demonstrated in the figure. The skew-t approximation performs extremely well in the plots as well. The approximated percentiles from skew-t are almost identical with the empirical percentiles for $m \geq 20$. The skew-t distributions are fit using method of moments with the exact theoretical moments computed. Apparently, the skew-t assumption is valid and we are
able to find an approximation under such assumption. Additionally, the computation for the skew-t distribution is quite simple and practically implemented. The function “st.cumulants.inversion” in the R package “sn” can perform the method of moments computation after we get the theoretical moments for $Y_m$.

Figure 4.10 allows comparison of the three approximation methods for the 1st percentile. The results are similar to results for the 5th percentile except the skew-t approximation is slightly liberal while the edgeworth approximation is slightly conservative; the CLT is unacceptably liberal. In conclusion, the skew-t approximation performs well when $m$ is small, and its computation is quite simple. As for Edgeworth series approximation, its performance is close to skew-t with four terms in the series. However, the approximating distribution function is not strictly a monotonically increasing distribution function because Edgeworth series uses a polynomial equation as an approximation. That makes the method unstable with multiple solutions for a specific $\alpha$ value cut-off. In Figure 4.9 and Figure 4.10, we are able to find the best solutions for Edgeworth series, because we know exactly where they should be. In practice, this instability makes it not preferable. Therefore, among the three different approximations of our null distribution, we suggest skew-t approximation when $m$ is “small”. Certainly, the approximation is needed only when $m$ is relatively small, since when $m \to \infty$, the true distribution of $Y_m$, as well as Edgeworth
series and skew-t approximation, converge to a normal distribution by CLT.

4.4.2 Power Study using Skew-t Approximated Null Distribution

So far, we have concluded that the skew-t distribution is a good approximation of the theoretical null distribution for testing (4.2) using a test based on the first PLS score of each GO term. We created a statistic

\[ Y_m = \frac{W_m - m\mu}{\sigma \sqrt{m}} , \]

which is a standardized version of \( W_m \), defined in Equation (4.13). This statistic is slightly different from the t statistics we created originally in Equation (4.5). To be exact, \( W_m \) can be defined as \( W_m = (\mathbf{\bar{S}}_0 - \mathbf{\bar{S}}_1)||X^T y|| \). To find out whether this modification can impact the validation of our statistic, we need to study its power using simulation.

For the next step, we simulate different alternative distributions and compute their powers. Recall our notation in Equation (4.1) and (4.2). The null hypothesis is \( H_0 : \mu_0 = \mu_1 \), where \( \mu_k = (\mu_{k1}, \ldots, \mu_{km})^T \). To simulate our alternatives, we gradually increase the distance between the elements of the mean vectors \( \mu_0 \) and \( \mu_1 \), where the mean difference of each simulated gene is de-
fined as $2\mu = \mu_0 - \mu_1$. In other words, we consider alternatives of the form

$$\mu_{0j} - \mu_{1j} = 2\mu$$

for $j = 1, \ldots, m$. For $\mu$, values $0, 0.1, 0.3, \cdots, 0.9$ are used. Each of these alternative distributions is simulated with 20000 data points. Then we plot the alternative distributions along with the approximated null distribution obtained from the skew-t distribution. Note that as $\mu = 0$, we are actually simulating the null distribution. Different values of $m$ are used, where $m$ is the number of genes related to each GO term.

The plots of the alternative distributions along with the skewed-t approximated null distribution are presented in Figure 4.11. First of all, we would like to see the simulated distribution very close to our theoretically approximated null as $\mu = 0$. Consistent with our previous results on the percentile estimation, the simulated null is a bit off from the theoretically approximated null as $m = 1, 2$. However, as $m > 3$, they are almost identical. This indicates that our approximation with skew-t performs rather well. Furthermore, with the increase of the mean difference, the alternatives are being separated from the null distribution, which suggests that our test statistics have fairly good powers.

Table 4.2 lists powers computed for different values of $m$ and $\mu$. By increasing the mean difference of all genes related to a GO term, the powers increase rapidly. Additionally, GO terms with larger $m$ have larger power when using
Figure 4.11: Compare alternative distribution with skew-t null distribution as $\mu$ increases with different values of $m$.

the same mean difference for all genes. It makes sense because a GO term with two differentially expressed genes should be more detectable than a GO term with one differentially expressed gene with the same significance level.
Table 4.2: Power calculation by changing mean difference $\mu$ of all genes related to each GO. $H_0 : \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a : \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\mu$</th>
<th>0</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.051</td>
<td>0.067</td>
<td>0.233</td>
<td>0.529</td>
<td>0.810</td>
<td>0.951</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.054</td>
<td>0.079</td>
<td>0.328</td>
<td>0.734</td>
<td>0.955</td>
<td>0.997</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.048</td>
<td>0.079</td>
<td>0.394</td>
<td>0.842</td>
<td>0.991</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.049</td>
<td>0.084</td>
<td>0.462</td>
<td>0.907</td>
<td>0.998</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.048</td>
<td>0.090</td>
<td>0.525</td>
<td>0.953</td>
<td>0.999</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.050</td>
<td>0.095</td>
<td>0.576</td>
<td>0.969</td>
<td>0.999</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.047</td>
<td>0.094</td>
<td>0.630</td>
<td>0.986</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.047</td>
<td>0.095</td>
<td>0.674</td>
<td>0.992</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.051</td>
<td>0.104</td>
<td>0.719</td>
<td>0.997</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.053</td>
<td>0.105</td>
<td>0.758</td>
<td>0.998</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Next, instead of changing the mean difference of all genes related to each GO term, we change only one gene for each GO term. In other words, we change only one of the elements for the mean vector $\mu_0$ and $\mu_1$, thus considering alternatives of the form $\mu_{01} - \mu_{11} = 2\mu$ and $\mu_{0j} - \mu_{1j} = 0$ for $j = 2, \ldots, m$. The other elements of the mean vectors stay as zero. The results are presented in Table 4.3. The power is not as good as the previous results. But since only one of the genes of each GO term is simulated as truly differentially expressed, we should have less power than before. It is also quite reasonable to see that when only one gene of each GO term is important, the power decreases when $m$ increases, because more non-differentially expressed genes are related to each
Table 4.3: Power calculation by changing mean difference $\mu$ of only one of the genes related to each GO. $H_0 : \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a : \mu_{01} - \mu_{11} = 2\mu$, and $\mu_{0j} = \mu_{1j} = 0$ for $j = 2, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20.

<table>
<thead>
<tr>
<th>$m \setminus \mu$</th>
<th>0</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.05(0.002)</td>
<td>0.070(0.002)</td>
<td>0.237(0.004)</td>
<td>0.536(0.004)</td>
<td>0.814(0.003)</td>
<td>0.951(0.002)</td>
</tr>
<tr>
<td>2</td>
<td>0.050(0.002)</td>
<td>0.062(0.002)</td>
<td>0.182(0.003)</td>
<td>0.426(0.004)</td>
<td>0.714(0.004)</td>
<td>0.909(0.002)</td>
</tr>
<tr>
<td>3</td>
<td>0.050(0.002)</td>
<td>0.059(0.002)</td>
<td>0.152(0.003)</td>
<td>0.368(0.004)</td>
<td>0.657(0.004)</td>
<td>0.873(0.003)</td>
</tr>
<tr>
<td>4</td>
<td>0.048(0.002)</td>
<td>0.058(0.002)</td>
<td>0.137(0.003)</td>
<td>0.332(0.004)</td>
<td>0.607(0.004)</td>
<td>0.843(0.003)</td>
</tr>
<tr>
<td>5</td>
<td>0.048(0.002)</td>
<td>0.053(0.002)</td>
<td>0.123(0.003)</td>
<td>0.302(0.004)</td>
<td>0.561(0.004)</td>
<td>0.800(0.003)</td>
</tr>
<tr>
<td>6</td>
<td>0.050(0.002)</td>
<td>0.058(0.002)</td>
<td>0.123(0.003)</td>
<td>0.291(0.004)</td>
<td>0.541(0.004)</td>
<td>0.791(0.004)</td>
</tr>
<tr>
<td>7</td>
<td>0.050(0.002)</td>
<td>0.057(0.002)</td>
<td>0.117(0.003)</td>
<td>0.270(0.004)</td>
<td>0.515(0.004)</td>
<td>0.768(0.004)</td>
</tr>
<tr>
<td>8</td>
<td>0.047(0.002)</td>
<td>0.051(0.002)</td>
<td>0.100(0.003)</td>
<td>0.258(0.004)</td>
<td>0.466(0.004)</td>
<td>0.722(0.004)</td>
</tr>
<tr>
<td>9</td>
<td>0.053(0.002)</td>
<td>0.060(0.002)</td>
<td>0.109(0.003)</td>
<td>0.241(0.004)</td>
<td>0.469(0.004)</td>
<td>0.719(0.004)</td>
</tr>
<tr>
<td>10</td>
<td>0.049(0.002)</td>
<td>0.053(0.002)</td>
<td>0.099(0.002)</td>
<td>0.221(0.004)</td>
<td>0.441(0.004)</td>
<td>0.692(0.004)</td>
</tr>
</tbody>
</table>

4.4.3 Simulation Study of our Test using Skew-t Approximated Null Distribution

The power estimations in the previous section appear to be reasonable using the test statistic $Y_m$ with a skew-t approximated null distribution. We create this test statistic trying to replace the t-test in our DEA-PLS procedure. Hence, our next step is to check whether this test is effective to find significant GO terms. We first are going to apply the test on the data sets we simulated in...
Section 3.3.1.

We have the exact same setup for the simulated data sets (see Section 3.3.1). We evaluate test statistics $Y_m$ for the GO terms in each data set and find the p-values using the null distribution approximated by a skew-t distribution. The p-values are then adjusted using the approach by Benjamini and Hochberg (1995). The results are presented in Figure 4.12, which is similar to Figure 3.2 with additional sensitivity and specificity results for the test using the skew-t distribution.

Figure 4.12: Sensitivity and specificity of four methods as a function of $\delta$ in the simulation study. Sensitivities are based on averages across all simulation replicates, with pointwise approximate 95% confidence intervals shown as vertical bars. Fisher’s exact test is solid line, DEA-PLS using t null is dashed line, DEA-Mean is dotted line, GSEA is dashed-dotted line and DEA-PLS using skew-t null is long-dashed line. $\alpha_l = 0.05$ for Fisher’s exact test. All GO-level p-values were BH-adjusted using $\alpha_G = 0.05$. 

132
The results using our proposed null distribution are performing as well as the DEA-PLS and much better than the other three approaches. This shows the effectiveness of our test statistics.

In our proposed DEA-PLS procedure (see Section 3.2), the first score of PLS is used in a two-sample equal-variance t test. There is no theoretical adjustment for the number of genes $m$ related to each GO term and how this might affect the p-value, but we know based on our theoretical derivation GO terms with a larger number of genes tend to be more significant using this simple t test. In the Chapter 3 version of DEA-PLS, the adjustment for GO terms with large number of $m$ is done using cross-validation to determine whether the first PLS latent variable significantly contributes to the prediction of $y$. If the first PLS component happens to over-fit $y$, we simply treat that gene set as non-important. Otherwise, we keep that component for further analysis. This filtering process limits the chances of retaining irrelevant latent summaries, and is most intense for GO terms that contain large number of genes. Though cross-validation seems to work well from our simulations studies, it is more of an ad hoc approach without any theoretical justification. Furthermore, it is computationally intensive to perform cross-validation for each GO term of interest. With our skew-t approximated null distribution, we are able to not only adjust our tests depending on the number of genes within each GO term,
but also significantly reduce the computation time.

In conclusion, by doing the theoretical derivation, we are able to find a test statistic that is more valid than the standard two-sample equal-variance t tests on the first PLS score. In the mean time, we find a theoretically justified adjustment for GO terms with larger number of genes $m$ and dramatically reduce the computation time. From our simulation study, we find that the new tests have good power and are effective at finding significant GO terms in our simulation studies. Hence, the “new” DEA-PLS procedure is an improvement over the old in that its theoretical underpinnings are more sure and it is much less computationally demanding. At the same time, no power has been lost and nominal significance levels are maintained.

## 4.5 Extension of Our Test Statistics for Unbalanced Case

We define our binary response vector as $y = \begin{bmatrix} 0_n^T \\ 1_n^T \end{bmatrix}$ and derived the null distribution assuming a balanced design with $n_0 = n_1 = n$ samples sizes for each class. If we allow an unbalanced design, i.e., $n_0 \neq n_1$, the derivation of the null distribution turns out to be quite similar. Assume we have the same notation as Section 4.1, except for the response vector, which becomes
\[ y = \begin{bmatrix} \mathbf{0}^T_{n_0} & \mathbf{1}^T_{n_1} \end{bmatrix} \]. Then Equation (4.5) is changed to

\[
\bar{S}_0 - \bar{S}_1 = \begin{bmatrix} \frac{1}{n_0} \mathbf{1}^T_{n_0} & -\frac{1}{n_1} \mathbf{1}^T_{n_1} \end{bmatrix} X w_1
\]

\[
= \begin{bmatrix} \frac{1}{n_0} \mathbf{1}^T_{n_0} & -\frac{1}{n_1} \mathbf{1}^T_{n_1} \end{bmatrix} \frac{XX^Ty}{||X^Ty||}
\]

\[
= \left( \begin{bmatrix} \frac{1}{n_0} \mathbf{1}^T_{n_0} & -\frac{1}{n_1} \mathbf{1}^T_{n_1} \end{bmatrix} X \right) \left( X^Ty \right) \frac{1}{||X^Ty||}
\]

\[
= \begin{bmatrix} n_1 \bar{X}_{1,1} \\ n_1 \bar{X}_{1,m} \end{bmatrix} \frac{1}{||X^Ty||} \tag{4.19}
\]

where \( \bar{X}_{k,j} \) is the mean expression level for gene \( j \) in treatment group \( y = k \).

Hence

\[
\frac{\sqrt{n_0}}{\sqrt{n_1}}(\bar{S}_0 - \bar{S}_1)||X^Ty|| = \sqrt{n_0 n_1} \sum_{j=1}^{m} (\bar{X}_{0,j} \bar{X}_{1,j} - \bar{X}_{1,j} \bar{X}_{1,j}) .
\]

Under the same distributional assumption in Equation (4.4), for \( H_0: \mu_1 = \)
\( \mu_2 = 0 \), then

\[
\begin{cases}
X_{0.1} = \frac{1}{n_0} \sum_{i=1}^{n_0} x_{0i1}, \cdots, X_{0.m} = \frac{1}{n_0} \sum_{i=1}^{n} x_{0im} \sim N(0, \frac{1}{n_0}) \\
X_{1.1} = \frac{1}{n_1} \sum_{i=1}^{n_1} x_{1i1}, \cdots, X_{1.m} = \frac{1}{n_1} \sum_{i=1}^{n} x_{1im} \sim N(0, \frac{1}{n_1})
\end{cases}
\]

\[
\Rightarrow \begin{cases}
\sqrt{n_0}X_{0.1}, \cdots, \sqrt{n_0}X_{0.m} \sim N(0, 1) \\
\sqrt{n_1}X_{1.1}, \cdots, \sqrt{n_1}X_{1.m} \sim N(0, 1)
\end{cases}
\]

\[
\Rightarrow \begin{cases}
\sqrt{n_0n_1}(X_{1.1} - X_{1.1}), \cdots, \sqrt{n_0n_1}(X_{1.m} - X_{1.m}) \sim \chi_1^2 \\
\sqrt{n_0n_1}(X_{0.1} - X_{1.1}), \cdots, \sqrt{n_0n_1}(X_{0.m} - X_{1.m}) \sim f_p
\end{cases}
\]

where \( f_p \) is the density of a product of two independent random variables distributed from \( N(0, 1) \). Therefore the new adjusted test statistics \( T_{adj} = \frac{\sqrt{n_0}}{\sqrt{n_1}}(S_0 - \overline{S}_1)||X^Ty|| \) follows the same distribution as \( W_m \) in Equation 4.8 for the balanced design.
Chapter 5

Impact of Test Statistics with Different Response Values

So far we have defined our binary response variable as \( y = \begin{bmatrix} 0^T_{n_0} & 1^T_{n_1} \end{bmatrix} \).

We are able to obtain a good approximation of the null distribution in Section 4. There has been some discussion in the literature about different results from PLS-DA analysis with different codings of the response variable \( y \) (Barker and Rayens, 2003). We will study the impact on our testing procedure with different codings of the response variable in this section.
5.1 Preliminary Results with Different Coding for Response Variables

We first keep all the assumptions from Section 4 and only recode the response variable as \( y = \begin{bmatrix} 1^T_n & -1^T_n \end{bmatrix} \) so that we are again in the case of a balanced design. It turns out that the null distribution becomes much simpler. Assume we have the same notation as Equation (4.1) except the response variable. Then Equation (4.5) is changed to

\[
\overline{S}_0 - \overline{S}_1 = \frac{1}{n} \begin{bmatrix} 1^T_n & -1^T_n \end{bmatrix} Xw_1
\]

\[
= \frac{1}{n} y'X'X'y
\]

\[
= \frac{1}{n} ||X'y||
\]

\[
= \frac{1}{n} \left\| \begin{bmatrix} n\overline{x}_{0.1} - n\overline{x}_{1.1} \\ \vdots \\ n\overline{x}_{0.m} - n\overline{x}_{1.m} \end{bmatrix} \right\|
\]

\[
= \frac{1}{n} \sqrt{(\overline{x}_{0.1} - \overline{x}_{1.1})^2 + \cdots + (\overline{x}_{0.m} - \overline{x}_{1.m})^2}
\] (5.1)
Similarly, for $H_0$, we have: $\mu_0 = \mu_1 = 0$, then

\[
\begin{align*}
\begin{cases}
\overline{X}_{0.1} &= \frac{1}{n} \sum_{i=1}^{n} x_{0i1}, \cdots, \overline{X}_{0.m} = \frac{1}{n} \sum_{i=1}^{n} x_{0im} \sim N(0, \frac{1}{n}) \\
\overline{X}_{1.1} &= \frac{1}{n} \sum_{i=1}^{n} x_{1i1}, \cdots, \overline{X}_{1.m} = \frac{1}{n} \sum_{i=1}^{n} x_{1im} \sim N(0, \frac{1}{n})
\end{cases}
\Rightarrow \overline{X}_{0.1} - \overline{X}_{1.1}, \cdots, \overline{X}_{0.m} - \overline{X}_{1.m} \sim N(0, \frac{2}{n})
\Rightarrow \overline{X}_{0.1} - \overline{X}_{1.1}, \cdots, \overline{X}_{0.m} - \overline{X}_{1.m} \sim \frac{\sqrt{2}}{\sqrt{n}}N(0, 1)
\end{align*}
\]

Hence,

\[
\begin{align*}
S_0 - S_1 &= \sqrt{\left(\overline{X}_{0.1} - \overline{X}_{1.1}\right)^2 + \cdots + \left(\overline{X}_{0.m} - \overline{X}_{1.m}\right)^2} \\
&\overset{d}{=} \sqrt{\left(\frac{\sqrt{2}}{\sqrt{n}}N(0, 1)\right)^2 + \cdots + \left(\frac{\sqrt{2}}{\sqrt{n}}N(0, 1)\right)^2} \\
&\overset{d}{=} \frac{\sqrt{2}}{\sqrt{n}}\sqrt{\chi^2_m} \quad (5.2)
\end{align*}
\]

For alternative $H_1$, we have $\mu_0 \neq \mu_1$, where $\mu_0 = \{\mu_j\}$ and $\mu_1 = -\{\mu_j\}$, $j = 1, \cdots, m$, then

\[
\begin{align*}
\begin{cases}
\overline{X}_{0.j} &= \frac{1}{n} \sum_{i=1}^{n} x_{0ij} \sim N(\mu_j, \frac{1}{n}) \quad , j = 1, \cdots, m \\
\overline{X}_{1.j} &= \frac{1}{n} \sum_{i=1}^{n} x_{1ij} \sim N(-\mu_j, \frac{1}{n})
\end{cases}
\Rightarrow \overline{X}_{0.j} - \overline{X}_{1.j} \sim N(2\mu_j, \frac{2}{n}), j = 1, \cdots, m \\
\Rightarrow \overline{X}_{0.j} - \overline{X}_{1.j} \sim \frac{\sqrt{2}}{\sqrt{n}}N(0, 1) + 2\mu_j, j = 1, \cdots, m.
\end{align*}
\]
Therefore,

\[
\overline{S}_0 - \overline{S}_1 = \sqrt{(X_{0,1} - X_{1,1})^2 + \cdots + (X_{0,m} - X_{1,m})^2}
\]

\[
\overset{d}{=} \sqrt{\left(\frac{\sqrt{2}}{\sqrt{n}} N(0,1) + 2\mu_1\right)^2 + \cdots + \left(\frac{\sqrt{2}}{\sqrt{n}} N(0,1) + 2\mu_m\right)^2}
\]

\[
= \sqrt{\frac{2}{n} \chi^2_m + \sum_{j=1}^{m} \frac{4\sqrt{2}}{\sqrt{n}} \mu_j N(0,1) + 4 \sum_{j=1}^{m} (\mu_j)^2}
\]

\[
(5.3)
\]

This is much simpler than the derivation in Section 4.2. We would like to find out what the difference is between the two type of tests when the only difference is how the response variable \( y \) is defined.

To see the difference we work out the details for \( m = 1, m = 2 \) and \( m = 3 \).

Recall for \( y = \begin{bmatrix} 0^T_n & 1^T_n \end{bmatrix} \),

\[
\overline{S}_0 - \overline{S}_1 = \begin{bmatrix} (n\overline{X}_{0,1} - n\overline{X}_{1,1}), \cdots, (n\overline{X}_{0,m} - n\overline{X}_{1,m}) \end{bmatrix} \begin{bmatrix} \overline{X}_{1,1} \\ \vdots \\ \overline{X}_{1,m} \end{bmatrix} \frac{1}{n||X^Ty||}
\]

Let \( \Delta S_1 = (\overline{S}_0 - \overline{S}_1)||X^Ty|| \) denote the test statistic of first PLS score for \( y = \begin{bmatrix} 0^T_n & 1^T_n \end{bmatrix} \) and \( \Delta S_2 = (\overline{S}_0 - \overline{S}_1)\sqrt{\frac{m}{2}} \) denotes the test statistic for
$y = \begin{bmatrix} 1^T_n & -1^T_n \end{bmatrix}$. Consequently,

$$
\Delta S_1 = [(\overline{X}_{0,1} - \overline{X}_{1,1}), \cdots, (\overline{X}_{0,m} - \overline{X}_{1,m})] \begin{bmatrix} n\overline{X}_{1,1} \\ \vdots \\ n\overline{X}_{1,m} \end{bmatrix} = n(\overline{X}_{0,1} - \overline{X}_{1,1})\overline{X}_{1,1} + \cdots + n(\overline{X}_{0,m} - \overline{X}_{1,m})\overline{X}_{1,m}
$$

and

$$(\Delta S_2)^2 = \frac{n}{2}(\overline{X}_{0,1} - \overline{X}_{1,1})^2 + \cdots + \frac{n}{2}(\overline{X}_{0,m} - \overline{X}_{1,m})^2$$

where $\Delta S_1$ is tested against approximated skew-t distribution and $(\Delta S_2)^2$ is tested against Chi-Square distribution with $m$ degrees of freedom.

Therefore, for $m = 1$ we are actually using the test statistics:

$$\Delta S_1 = n(\overline{X}_{0,1} - \overline{X}_{1,1})\overline{X}_{1,1}$$

$$(\Delta S_2)^2 = \frac{n}{2}(\overline{X}_{0,1} - \overline{X}_{1,1})^2.$$ 

For $m=2$:

$$\Delta S_1 = n(\overline{X}_{0,1} - \overline{X}_{1,1})\overline{X}_{1,1} + n(\overline{X}_{0,2} - \overline{X}_{1,2})\overline{X}_{1,2}$$

$$(\Delta S_2)^2 = \frac{n}{2}(\overline{X}_{0,1} - \overline{X}_{1,1})^2 + \frac{n}{2}(\overline{X}_{0,2} - \overline{X}_{1,2})^2.$$
For $m=3$:

$$\Delta S_1 = n(\overline{X}_{0.1} - \overline{X}_{1.1})\overline{X}_{1.1} + n(\overline{X}_{0.2} - \overline{X}_{1.2})\overline{X}_{1.2} + n(\overline{X}_{0.3} - \overline{X}_{1.3})\overline{X}_{1.3}$$

$$\left(\Delta S_2\right)^2 = \frac{n}{2}(\overline{X}_{0.1} - \overline{X}_{1.1})^2 + \frac{n}{2}(\overline{X}_{0.2} - \overline{X}_{1.2})^2 + \frac{n}{2}(\overline{X}_{0.3} - \overline{X}_{1.3})^2.$$ 

Assume that $\overline{X}_{0.1} = a, \overline{X}_{1.1} = -a, \overline{X}_{0.2} = b, \overline{X}_{1.2} = -b, \overline{X}_{0.3} = c, \overline{X}_{1.3} = -c.$

For $m = 1$, 

$$\Delta S_1 = -2na^2$$

$$\Delta S_2 = 2na^2.$$ 

For $m = 2$, 

$$\Delta S_1 = n(a + a)(-a) + n(b + b)(-b) = -2n(a^2 + b^2)$$

$$\left(\Delta S_2\right)^2 = \frac{n}{2}(a + a)^2 + \frac{n}{2}(b + b)^2 = 2n(a^2 + b^2)$$

For $m = 3$, 

$$\Delta S_1 = n(a + a)(-a) + n(b + b)(-b) + n(c + c)(-c) = -2n(a^2 + b^2 + c^2)$$

$$\left(\Delta S_2\right)^2 = \frac{n}{2}(a + a)^2 + \frac{n}{2}(b + b)^2 + \frac{n}{2}(c + c)^2 = 2n(a^2 + b^2 + c^2)$$
Derivation of the first PLS score under the two different codings shows that the two statistics are essentially the same with only a sign difference when expressions are centered about the mean for a given gene.

To study more about the test statistics with response variable \( y = [1^T_n -1^T_n] \), we take a closer look at the power. Note that a test based on the new statistic is tested using its own derived null distribution which is the square root of a chi square distribution. The study is similar to the one in Section 4.4.2. The powers of the tests are listed in Table 5.1 for different values of \( m \) and \( \mu \) for all genes in the group, and Table 5.2 lists the powers of the tests for different values of \( m \) and \( \mu \) for only one of the genes in the group. In general, the powers of the tests using response variable \( y = [1^T_n -1^T_n] \) follow the same trend as using response variable \( y = [0^T_n 1^T_n] \). However, the powers are slightly larger for response variable \( y = [1^T_n -1^T_n] \). We take a closer look at the critical values of the two statistics. Since the statistic \( \Delta S_1 \) is standardized, we recalculate the non-standardized version for comparison purpose. The \( \alpha = 0.05 \) critical values are listed in Table 5.3. From our previous calculation, we find out that both test statistics are essentially the same except for a sign difference. We can also see from the critical values that they are closely related but not exactly the same. We will show they can be equivalent with some restriction on our assumption in the next Section.
Table 5.1: Power calculation by changing mean difference $\mu$ of all genes related to each GO with response variable $y = [1, -1]$. $H_0 : \mu_{0j} = \mu_{1j}$, $j = 1, \ldots, m$ versus $H_a : \mu_{0j} - \mu_{1j} = 2\mu$, $j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20.

<table>
<thead>
<tr>
<th>$m \backslash \mu$</th>
<th>0</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.048(0.002)</td>
<td>0.071(0.003)</td>
<td>0.261(0.004)</td>
<td>0.6(0.005)</td>
<td>0.875(0.003)</td>
<td>0.982(0.001)</td>
</tr>
<tr>
<td>2</td>
<td>0.052(0.002)</td>
<td>0.082(0.003)</td>
<td>0.377(0.005)</td>
<td>0.818(0.004)</td>
<td>0.983(0.001)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>3</td>
<td>0.047(0.002)</td>
<td>0.083(0.003)</td>
<td>0.466(0.005)</td>
<td>0.919(0.003)</td>
<td>0.998(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>4</td>
<td>0.051(0.002)</td>
<td>0.093(0.003)</td>
<td>0.549(0.005)</td>
<td>0.962(0.002)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>5</td>
<td>0.051(0.002)</td>
<td>0.102(0.003)</td>
<td>0.625(0.005)</td>
<td>0.984(0.001)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>6</td>
<td>0.053(0.002)</td>
<td>0.106(0.003)</td>
<td>0.676(0.005)</td>
<td>0.993(0.001)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>7</td>
<td>0.05(0.002)</td>
<td>0.11(0.003)</td>
<td>0.738(0.004)</td>
<td>0.997(0.001)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>8</td>
<td>0.052(0.002)</td>
<td>0.111(0.003)</td>
<td>0.778(0.004)</td>
<td>0.999(0.000)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>9</td>
<td>0.053(0.002)</td>
<td>0.118(0.003)</td>
<td>0.817(0.004)</td>
<td>0.999(0.000)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
<tr>
<td>10</td>
<td>0.047(0.002)</td>
<td>0.121(0.003)</td>
<td>0.849(0.004)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
<td>1(0.000)</td>
</tr>
</tbody>
</table>

5.2 Relations between Different Coding of Response Variable

From the preliminary results, we are able to see a lot of connections between the two types of codings $y = \begin{bmatrix} 0^T_n & 1^T_n \end{bmatrix}$ and $y = \begin{bmatrix} 1^T_n & -1^T_n \end{bmatrix}$. Even though the null distributions appear to be different, the tests have a lot in common. The powers from two tests are very similar and for some special cases, the test statistics computed are equivalent. In this section we formalize this relation.

We keep our notation from Section 4. Let $x_{kij}$ represent expression level
Table 5.2: Power calculation by changing mean difference $\mu$ of only one genes related to each GO with response variable $y = [1, -1]$. $H_0: \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a: \mu_{01} - \mu_{11} = 2\mu$, and $\mu_{0j} = \mu_{1j} = 0$ for $j = 2, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20.

<table>
<thead>
<tr>
<th>$m \setminus \mu$</th>
<th>0</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.048(0.002)</td>
<td>0.072(0.003)</td>
<td>0.266(0.004)</td>
<td>0.607(0.005)</td>
<td>0.879(0.003)</td>
<td>0.982(0.001)</td>
</tr>
<tr>
<td>2</td>
<td>0.048(0.002)</td>
<td>0.066(0.002)</td>
<td>0.209(0.004)</td>
<td>0.495(0.005)</td>
<td>0.796(0.004)</td>
<td>0.956(0.002)</td>
</tr>
<tr>
<td>3</td>
<td>0.048(0.002)</td>
<td>0.060(0.002)</td>
<td>0.173(0.004)</td>
<td>0.439(0.005)</td>
<td>0.754(0.004)</td>
<td>0.935(0.002)</td>
</tr>
<tr>
<td>4</td>
<td>0.047(0.002)</td>
<td>0.055(0.002)</td>
<td>0.154(0.004)</td>
<td>0.398(0.005)</td>
<td>0.707(0.005)</td>
<td>0.912(0.003)</td>
</tr>
<tr>
<td>5</td>
<td>0.048(0.002)</td>
<td>0.058(0.002)</td>
<td>0.143(0.004)</td>
<td>0.367(0.005)</td>
<td>0.665(0.005)</td>
<td>0.890(0.003)</td>
</tr>
<tr>
<td>6</td>
<td>0.053(0.002)</td>
<td>0.063(0.002)</td>
<td>0.141(0.003)</td>
<td>0.349(0.005)</td>
<td>0.643(0.005)</td>
<td>0.876(0.003)</td>
</tr>
<tr>
<td>7</td>
<td>0.050(0.002)</td>
<td>0.057(0.002)</td>
<td>0.131(0.003)</td>
<td>0.327(0.005)</td>
<td>0.609(0.005)</td>
<td>0.860(0.003)</td>
</tr>
<tr>
<td>8</td>
<td>0.049(0.002)</td>
<td>0.054(0.002)</td>
<td>0.115(0.003)</td>
<td>0.294(0.005)</td>
<td>0.565(0.005)</td>
<td>0.827(0.004)</td>
</tr>
<tr>
<td>9</td>
<td>0.057(0.002)</td>
<td>0.064(0.002)</td>
<td>0.122(0.003)</td>
<td>0.290(0.005)</td>
<td>0.562(0.005)</td>
<td>0.815(0.004)</td>
</tr>
<tr>
<td>10</td>
<td>0.048(0.002)</td>
<td>0.055(0.002)</td>
<td>0.107(0.003)</td>
<td>0.261(0.004)</td>
<td>0.530(0.005)</td>
<td>0.792(0.004)</td>
</tr>
</tbody>
</table>

for gene $j$ corresponding to individual $i$ within treatment group $y = k$, where $k = 0, 1, i = 1, 2, \ldots, n$ and $j = 1, \ldots, m$. Consequently, the design matrix for the linear model is

$$X = \begin{bmatrix} X_0 \\ X_1 \end{bmatrix}, \quad X_k = \{x_{ki}j\}_{i=1,2,\ldots,n,j=1,\ldots,m}, \text{ for } k = 0, 1.$$ (5.4)

There are a total of $2n$ individuals or samples and $m$ genes. Assuming an equal number of cases and controls so that our response variable $y$ is a $2n \times 1$ vector,
Table 5.3: The comparison of the $p = 0.05$ cutoffs for different response variables. $\Delta S_1$ denotes the statistic for variable $y = [0, 1]$. $(\Delta S_2)^2$ denotes the statistic for variable $y = [1, -1]$

<table>
<thead>
<tr>
<th>m</th>
<th>$\Delta S_1$</th>
<th>$(\Delta S_2)^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-4.3283</td>
<td>3.8415</td>
</tr>
<tr>
<td>2</td>
<td>-6.7312</td>
<td>5.9915</td>
</tr>
<tr>
<td>3</td>
<td>-8.7929</td>
<td>7.8147</td>
</tr>
<tr>
<td>4</td>
<td>-10.6814</td>
<td>9.4877</td>
</tr>
<tr>
<td>5</td>
<td>-12.4636</td>
<td>11.0705</td>
</tr>
<tr>
<td>6</td>
<td>-14.0533</td>
<td>12.5916</td>
</tr>
<tr>
<td>7</td>
<td>-15.6017</td>
<td>14.0671</td>
</tr>
<tr>
<td>8</td>
<td>-17.1624</td>
<td>15.5073</td>
</tr>
<tr>
<td>9</td>
<td>-18.5956</td>
<td>16.9190</td>
</tr>
<tr>
<td>10</td>
<td>-20.0518</td>
<td>18.3070</td>
</tr>
</tbody>
</table>

the first PLS score is defined as

$$S = \frac{XX^Ty}{||X^Ty||}.$$  

Suppose we choose to define DEA-PLS test statistic as

$$T \equiv \frac{1}{n}(\overline{S}_0 - \overline{S}_1)||X^Ty|| = \frac{1}{n^2} \left[ \begin{array}{cc} 1^T & -1^T \end{array} \right] XX^Ty$$  \hspace{1cm} (5.5)

where $\overline{S}_0$, $\overline{S}_1$ are the means of first PLS score according to groups. The test
statistics obtained from different codings follows:

\[
y = \begin{bmatrix} 0_n \\ 1_n \end{bmatrix} \Rightarrow T_{[0,1]} = \sum_{j=1}^{m} (\overline{x}_{0,j} - \overline{x}_{1,j}) \overline{x}_{1,j},
\]

\[
y = \begin{bmatrix} -1_n \\ 0_n \end{bmatrix} \Rightarrow T_{[-1,0]} = \sum_{j=1}^{m} - (\overline{x}_{0,j} - \overline{x}_{1,j}) \overline{x}_{0,j},
\]

\[
y = \begin{bmatrix} 1_n \\ -1_n \end{bmatrix} \Rightarrow T_{[1,-1]} = \sum_{j=1}^{m} (\overline{x}_{0,j} - \overline{x}_{1,j})(\overline{x}_{0,j} - \overline{x}_{1,j}),
\]

\[
y = \begin{bmatrix} -1_n \\ 1_n \end{bmatrix} \Rightarrow T_{[-1,-1]} = \sum_{j=1}^{m} (\overline{x}_{0,j} - \overline{x}_{1,j})(-\overline{x}_{0,j} + \overline{x}_{1,j}). \tag{5.6}
\]

We can see this interesting relationship among these statistics: \( T_{[-1,1]} = - T_{[1,-1]} \) and \( T_{[-1,1]} = T_{[0,1]} + T_{[-1,0]} \). Both relations are very intuitive. When we reverse coding from \( y = [-1, 1] \) to \( y = [1, -1] \), the test statistics are able to capture the same absolute value of the differences between two classes. The second equation shows that using the coding of \( y = [0, 1] \) and \( y = [-1, 0] \), the sum of these two test statistics is equivalent to the test statistics with coding of \( y = [-1, 1] \).
5.3 Generalization of the Test Statistics

We found some interesting relationships between the test statistics with different coding: $T_{[-1,1]} = -T_{[1,-1]}$ and $T_{[-1,1]} = T_{[0,1]} + T_{[-1,0]}$. But there is no equivalence between $T_{[-1,1]}$ and $T_{[0,1]}$. According to the preliminary results, the two test statistics are equivalent when the expression means of two classes are centered around zero. At this point, we hope that the centering of the gene expressions can make the two test statistics equivalent. We have been assuming that the gene expressions in the data are already centered and scaled to variance of 1 by normalization. We would like to generalize the test statistics by including the normalization step in our inference, and along the way hope to find the equivalence of the test statistics with different codings.

5.3.1 New Notation

We need to redefine our notation. Let $w_{ijk}$ represent expression level for gene $j$ corresponding to individual $i$ within treatment $k$. In the case that we have balanced sample sizes, $n_0 = n$ and $n_1 = n$ for binary classes, we assume
the \( j \)th gene expression

\[
\mathbf{w}_{.,j} \overset{\text{indep}}{\sim} N(\beta_j, \sigma_j^2 \mathbf{I}), \beta_j = \begin{bmatrix} \mu_{0j} \mathbf{1}_n \\ \mu_{1j} \mathbf{1}_n \end{bmatrix}, j = 1, \ldots, m,
\]

where the first \( n_0 \) elements of \( \mathbf{w}_{.,j} \) are from one class and the latter \( n_1 \) elements are from the other class, i.e.

\[
\mathbf{w}_{.,j} = (w_{01j}, \ldots, w_{0n_0j}, w_{11j}, \ldots, w_{1n_1j})^T.
\]

We define hypotheses as

\[
H_0 : \mu_0 = \mu_1, \quad \text{versus} \quad H_a : \mu_0 \neq \mu_1,
\]

where \( \mu_0, \mu_1 \) are \( m \times 1 \) mean vectors. Notice the difference between Equation (5.7) and Equation (4.2), the new hypothesis does not limit the mean of the gene expression to be zero under the null.

We center and scale our expression level by first subtracting its mean then dividing by the pooled sample standard error,

\[
\mathbf{x}_{.,j} = \frac{\mathbf{w}_{.,j} - \mathbf{w}_{.,j} \mathbf{1}_{2n}}{\hat{\sigma}_j} = (I - P) \frac{\mathbf{w}_{.,j}}{\hat{\sigma}_j},
\]

(5.8)
where $P = \frac{1}{2n}1_{2n}1_{2n}^T$ is a $(2n) \times (2n)$ projection matrix and hence $I - P$ is also a projection matrix. The pooled sample standard error is defined as

$$\hat{\sigma}_j = \sqrt{\frac{(n_0 - 1)s_0^2 + (n_1 - 1)s_1^2}{n_0 + n_1 - 2} \left( \frac{1}{n_0} + \frac{1}{n_1} \right)}$$

(5.9)

where $s_k^2$ is the sample variance of $w_{k1j}, \ldots, w_{k_{nk}j}$ for class $k$.

The first score of the PLS procedure is then created based on the centered and scaled gene expressions $X$ obtained from Equation (5.8),

$$S = \frac{XX^Ty}{\|X^Ty\|}$$

and we still define the test statistic,

$$T \equiv \frac{1}{n}(\bar{S}_0 - \bar{S}_1)||X^Ty||$$

(5.10)

where $\bar{S}_0$, $\bar{S}_1$ are the means of first PLS score according to groups.

5.3.2 Equivalence of Test Statistics with Different Codings

With the new notation, Equation (5.6) still holds true using the centered and scaled data for different coding of the response variable. We first look
at sample means of $X$ within groups. Define $\overline{X}_{k,j}$ as the sample mean of the centered and scaled expression for gene $j$ in group $k$, then

\[
\overline{X}_{0,j} = \frac{1}{n} \left[ \begin{array}{cc} 1_T^n & 0^n_T \\ \end{array} \right] x_{.j} \\
= \frac{1}{n} \left[ \begin{array}{cc} 1_T^n & 0^n_T \\ \end{array} \right] (I - P) w_{.j} \frac{1}{\sigma_j} \\
= \frac{1}{2n} \left[ \begin{array}{cc} 1_T^n & -1_T^n \\ \end{array} \right] w_{.j} \frac{1}{\sigma_j} \\
= \frac{1}{2} (W_{0,j} - W_{1,j}) \frac{1}{\sigma_j},
\]

Similarly,

\[
\overline{X}_{1,j} = -\frac{1}{2} (W_{0,j} - W_{1,j}) \frac{1}{\sigma_j}.
\]

Hence the constraint leads to $\overline{X}_{1,j} = -\overline{X}_{0,j}$. As a result from Equation (5.6), we get $T_{[0,1]} = \sum_{j=1}^m 2\overline{X}_{0,j}(-\overline{X}_{0,j}), T_{[-1,0]} = \sum_{j=1}^m -2\overline{X}_{0,j}(\overline{X}_{0,j}) = T_{[0,1]}, T_{[-1,1]} = \sum_{j=1}^m 2\overline{X}_{0,j}(-2\overline{X}_{0,j}) = 2T_{[0,1]} = -T_{[1,-1]}$. Therefore, after centering and scaling, the test statistics for coding $y = [1, -1]$ and $y = [0, 1]$ are equal up to a constant multiplier of 2 and are therefore equivalent.
5.3.3 Generalized Test Statistics with Response Variable $y = [1, -1]$

Now we know that the test statistics are equivalent for different codings of $y$ after centering and scaling, we can just focus on one of them to extend the test for the generalized case.

**Derivation of the Test Statistics**

We focus on derivation of the test statistics with response variable $y = [1, -1]$.

We already know that

$$X_{0,j} = \frac{1}{2} \left( W_{0,j} - W_{1,j} \right) \frac{1}{\hat{\sigma}_j},$$

and,

$$X_{1,j} = -\frac{1}{2} \left( W_{0,j} - W_{1,j} \right) \frac{1}{\hat{\sigma}_j}.$$  

We define the test statistic for $H_0 : \mu_{0j} = \mu_{1j}$ as Equation (5.10):

$$T_{[1,-1]} = \sum_{j=1}^{m} \left( X_{0,j} - X_{1,j} \right)^2 = \sum_{j=1}^{m} \left( \frac{W_{0,j} - W_{1,j}}{\hat{\sigma}_j} \right)^2.$$  \hspace{1cm} (5.11)
Recall that

\[ w_{01j}, \ldots, w_{0n_0j} \overset{iid}{\sim} N(\mu_{0j}, \sigma_j^2), \]

\[ w_{11j}, \ldots, w_{1n_1j} \overset{iid}{\sim} N(\mu_{1j}, \sigma_j^2). \]

Under \( H_0 : \mu_{0j} = \mu_{1j} \), \( \frac{\bar{W}_{0j} - \bar{W}_{1j}}{\hat{\sigma}_j} \) is the test statistic from the pooled variance 2-sample t-test with degrees of freedom \( n_0 + n_1 - 2 \) where variances are assumed equal for the two classes. \( \left( \frac{\bar{W}_{0j} - \bar{W}_{1j}}{\hat{\sigma}_j} \right)^2 \) then follows an \( F \) distribution \( F_{\nu_1, \nu_2} \) with degrees of freedom \( \nu_1 = 1, \nu_2 = n_0 + n_1 - 2 \). Therefore under the null \( H_0 \)

\[ T_{[1,-1]} = \frac{d}{m} \sum_{j=1}^{m} F_{1, n_0 + n_1 - 2}. \]

\( T_{[1,-1]} \) is a sum of identically distributed \( F_{\nu_1, \nu_2} \). We can find the first four moments of \( F_{\nu_1, \nu_2} \) and approximate its null distribution using Edgeworth series expansion or skew-t distribution. For now, we assume independence between genes and thus independence between the \( F \)-distributed random variables. We will discuss the validity of this assumption later.

### Approximation of the Null Distribution

The null distribution we have on hand is very similar to the null distribution in Equation (4.13) with the only difference that we have a sum of identically distributed \( F_{\nu_1, \nu_2} \) instead of \( W_1 \). We hence are able to use the same approach for
approximating $W^m$ by using Edgeworth expansion or skew-t null distribution as in Chapter 4.

We need to obtain the first four moments of $F_{\nu_1,\nu_2}$ before we proceed to use Edgeworth Expansion or skew-t approximation. We compute the moments using the following equation:

$$m_z = \frac{\Gamma(\frac{\nu_1+2z}{2})\Gamma(\frac{\nu_1-2z}{2})}{\Gamma(\frac{\nu_1}{2})\Gamma(\frac{\nu_2}{2})} \left(\frac{\nu_2}{\nu_1}\right)^z,$$

where $z$ is the order of the moments, $\nu_1 = 1$ and $\nu_2 = n_0 + n_1 - 2$. The mean and variance of the distribution is given as $\mu = m_1$, $\sigma^2 = m_2 - m_1^2$. The first four cumulants for $F_{\nu_1,\nu_2}$ are then computed using a recursive function,

$$\begin{align*}
\kappa_1 &= m_1 \\
\kappa_2 &= m_2 - \kappa_1^2 \\
\kappa_3 &= m_3 - 3\kappa_2\kappa_1 - \kappa_1^3 \\
\kappa_4 &= m_4 - 4\kappa_3\kappa_1 - 3\kappa_2^2 - 6\kappa_2\kappa_1^2 - \kappa_1^4
\end{align*}$$

After finding the cumulants for $F_{\nu_1,\nu_2}$, we can use the three methods–central limit theorem, first order edgeworth expansion, skew-t by methods of moments–described in Section 4.3 to approximate the null distribution of $T_{[1,-1]}$. We briefly review the three approximating methods here.
By the CLT, as \( m \to \infty \), we approximately have

\[
\frac{\sqrt{m} \left( \frac{1}{m} T_{[1,-1]} - \mu \right)}{\sigma} \xrightarrow{d} N(0, 1) \Rightarrow T_{[1,-1]} = \sum_{i=1}^{m} F_{\nu_1, \nu_2} \approx N(m\mu, m\sigma^2)
\]

where \( \mu \) and \( \sigma^2 \) are the mean and variance of the \( F_{\nu_1, \nu_2} \) distribution.

For edgeworth expansion methods, we work with the standardized version of our test statistic \( \frac{T_{[1,-1]} - m\mu}{\sigma\sqrt{m}} \) that follows a density function which may be approximated to the first order by:

\[
f(T_{[1,-1]; x}) = \phi(x) \left( 1 + \frac{\kappa_3}{6\sqrt{m}\sigma^3} H_3(x) + \frac{\kappa_4}{24m\sigma^4} \frac{H_4(x)}{\sqrt{12m\sigma^6}} \right) + O(m^{-\frac{3}{2}}),
\]

where \( H_j(x) \) is the \( j \)th Hermite polynomial, defined by \( H_r(x)\phi(x) = (-1)^r d^r \phi(x)/dx^r \).

For Skew-t approximation, we compute the cumulants for \( \frac{T_{[1,-1]} - m\mu}{\sigma\sqrt{m}} \) using the cumulants found for \( F_{\nu_1, \nu_2} \) and obtain:

\[
K'(Y_m; t = 0) = t + \frac{\kappa_3}{2\sqrt{m}\sigma^3} t^2 + \frac{\kappa_4}{6m\sigma^4} t^3 + \cdots \big|_{t=0} = 0
\]

\[
K''(Y_m; t = 0) = 1 + \frac{\kappa_3}{\sqrt{m}\sigma^3} t + \frac{\kappa_4}{2m\sigma^4} t^2 + \cdots \big|_{t=0} = 1
\]

\[
K'''(Y_m; t = 0) = \frac{\kappa_3}{\sqrt{m}\sigma^3} + \frac{\kappa_4}{m\sigma^4} t + \cdots \big|_{t=0} = \frac{\kappa_3}{\sigma^3\sqrt{m}}
\]

\[
K^{(4)}(Y_m; t = 0) = \frac{\kappa_4}{m\sigma^4} + \cdots \big|_{t=0} = \frac{\kappa_4}{\sigma^4m}.
\]

Method of moments is used to solve for the four parameters of the skew-t
distribution. The resulting skew-t distribution is used as our null distribution.

We compare the three approximating methods using methods similar to those described in Section 4.4.1. To assess which of these methods have the best approximation, we compare them to empirical null distributions obtained from simulation. To generate the empirical null distribution of our test statistics, we first generate \( m \) gene expressions from the normal distribution \( N(4, 4) \) with sample size of 100. We choose mean = 4 and variance = 4, instead of mean = 0 and variance = 1 because we now have included the centering and scaling process in the computation of the test statistics. Each of these genes are generated from one normal distribution instead of a mixture of normal because under the null these genes are not differentially expressed. The \( m \) genes are being simulated independently. We compute the test statistic \( T_{[1,-1]} \) for this gene set to get one sample for the empirical null distribution of the test statistic. We then repeat this process, simulating 20000 sets of \( m \) genes and compute the test statistic for each gene set, to obtain the empirical null distributions. Different \( m \) is used for each empirical null distribution to reflect different size of gene sets.

We also find the three approximated distributions from our theoretical derivation using the methods we described earlier. Since we care more about the tail area of the distribution for testing purpose, we only compare the 95th
Figure 5.1: Estimation of 95th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 95th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points.

and 99th percentile of these three approximated distributions with those of the empirical distribution. The results are presented in Figure 5.1 and Figure 5.2, and are consistent with those in Section 4.4.1. The approximation by CLT is not very good when $m$ is small. Edgeworth expansion and skew-t approximation are performing much better, but Edgeworth approximation is not very stable due to the nature of using a polynomial function to approximate a distribution function because the polynomial function is not monotonically increasing and hence is not strictly a distribution function. We thus choose
Figure 5.2: Estimation of 99th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 99th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points.

skew-t as the optimal approximated null distribution.

**Simulation Studies**

We have derived the distribution of the test statistics under null and found its approximation using skew-t approximation. Now we need to compare our new test procedure with other methods with simulation. We compare this new test procedure with two popular methods: GSEA (Subramanian *et al.*, 2005) and PAGE (Kim and Volsky, 2005).
We described PAGE and GSEA in Section 2.4.2 and Section 2.4.3. We directly use the software available at http://www.broad.mit.edu/gsea/ for GSEA. Their software is programmed in Java. We also programmed our own version of PAGE using R. In the PAGE procedure, the statistic we use for each gene is the two-sample pooled variance t test. These statistics are then averaged across all genes within each GO term:

\[ T = \sum_{i=1}^{m} t_i, \]

where \( t_i \) is the two-sample student t statistic for gene \( i \). The null distribution is derived as a Z score via Central Limit Theorem:

\[ \sqrt{m} \left( \frac{1}{m} T - \mu \right) \frac{\sigma}{\sqrt{\sigma}} \approx N(0,1), \]

where \( \mu \) and \( \sigma^2 \) are the theoretical mean and variance of the student t distribution.

It is interesting to notice that this test statistic is similar to our new test statistic approximated using CLT. The difference is that the t statistic is used for each gene instead of the F statistic. Since PAGE uses CLT to find the null distribution, it does not work well when \( m \) is small. Kim and Volsky (2005) pointed out that drawback in the paper and suggested ignoring GO terms with
small number of genes. During the process of deriving our own test statistics, we have found that the convergence to Normal distribution by CLT is very slow and throwing away GO term with relatively small number of genes can prevent us finding useful GO terms. Thus other approximating methods were proposed and found to be performing better, i.e. skew-t approximation via methods of moments. Our approximating methods can be easily implemented to improve PAGE. From another perspective, we can regard our new test statistic $T_{[1,-1]}$ as improved version (better approximated null distribution when $m$ is small) of PAGE using F statistic for each individual genes even though we started our analysis using PLS procedure.

Our simulation setups are the same as Section 3.3.1. We simulate 3666 gene expressions with 16 of them differentially expressed. These genes are mapped to 566 GO terms from the molecular function hierarchy with 9 of them related to differentially expressed genes (See Section 3.3.1 for details). Various mean differences of the genes $2\delta$ are used with different $\delta$ value, 0, 0.1, 0.3, 0.5, 0.7, 0.9. We compare the sensitivity and specificity of six methods, Fisher’s exact test, DEA-PLS (t test version with cross validation described in Liu et al. (2007)), DEA-MEAN (Mean summary with t test described in Section 3, GSEA (software available from http://www.broad.mit.edu/gsea/), PAGE (average of t statistics using CLT for null distribution) and the new
test statistic we derived—the “new” DEA-PLS. The results are presented in Figure 5.3.

Figure 5.3: Sensitivity and specificity of four methods as a function of δ in the simulation study. Sensitivities are based on averages across all simulation replicates, with pointwise approximate 95% confidence intervals shown as vertical bars. Fisher’s exact test is solid line, DEA-PLS using t null is dashed line, DEA-Mean is dotted line, GSEA is dashed-dotted line PAGE is skew-t null is long-dashed line and the new derived test is short-long-dashed lines. α_I = 0.05 for Fisher’s exact test. All GO-level p-values were BH-adjusted using α_G = 0.05.

The GSEA results in Figure 3.2 are obtained from a program we implemented according to Mootha et al. (2003). To avoid possibly inaccurate implementation, here we use the software available online to perform the GSEA procedure. The software obtained very similar results in terms of true find-
ings with sensitivity at most 30% even for large mean differences. However, the specificity decreases a lot using GSEA via the software, meaning increased number of false positives among findings. The specificity is constantly around 97% so it isn’t shown on the specificity plot.

The main goal of our simulation is to see how PAGE and the new test statistic perform against other methods. PAGE seems to pick up signals more slowly than other methods with the increase of the mean difference. However, it has the similar sensitivity around 80% to DEA-mean and Fisher’s exact test once $\delta$ increases to 0.7. It is encouraging to see that our new test with skew-t approximated null is performing by far better than any other methods. It is able to find 100% of the important GO terms when $\delta$ is 0.7 and 0.9. This level of sensitivity has not been reached by any other methods. Looking at the specificity, we already pointed out that GSEA had the most false positives (not shown on the plot). For our new test statistics, we do not sacrifice too much specificity. The decreasing of the specificity is within one standard error of most other methods. We also see that PAGE has a relative high specificity even though its sensitivity increases slower than other methods with the increase of mean differences. So for cases that we strictly do not want large numbers of false findings even if we sacrifice some true findings, PAGE is recommended according to the simulations results.
Summary

Prior to this section, we found that in a more general setting where

\[ w_j \overset{iid}{\sim} N(\beta_j, \sigma_j^2 I), \beta_j = \begin{bmatrix} \mu_{0j}1_n \\ \mu_{1j}1_n \end{bmatrix}, j = 1, \ldots, m, \]

the test statistics are equivalent via centering and scaling procedure for different coding of the response variable \( y = [1, -1] \) and \( y = [0, 1] \). We thus proceed to pursue a test for response variable \( y = [1, -1] \). Because we incorporate the centering and scaling procedure, we are able to create a test that are applicable for more general assumptions, which is \( \mu_{0j} = \mu_{1j} \) instead of \( \mu_{0j} = \mu_{1j} = 0 \) from Section 4, and the variance of our gene expressions is not restricted to 1 anymore. We still have our assumption of independence between genes, which is unpractical in most cases. We will discuss the impact of having correlation later.

In this section, we are able to accomplish derivation of the new test statistics, finding the better approximation distribution of the statistic under null and compare the effectiveness of our new test statistic with other competitive methods (especially the most popular GSEA and PAGE, DEA-PLS with two sample t test). The results shows favor to new testing procedure because we have the best sensitivity while not sacrificing too much on specificity. We focus
on the derivation of the test statistics with balanced classes, \( n_0 = n_1 = n \). See Appendix E for the derivation of the test statistic with unbalanced cases.

5.4 Impact on the New Testing Procedure with Correlation between Genes

So far, we have derived all of our testing procedure based on the assumption that all genes related to each GO term are independent. This is not practical in some cases. Figure 5.4 is an example of correlations between two genes related to a GO term for an observed experiment data set. The samples have a binary class response variable, diseased and non-diseased. We can clearly see the correlation between the two genes. Furthermore, the disease status seems to have an impact on the correlation. These correlation relations are expected to be stronger for genes related to one GO term if the GO mappings to genes are accurate.

The goal of this section of the dissertation is not to find a test that can be applied to correlated gene sets. That will be the next step of our research. We just want to see how much the correlation can impact our test statistics derived under the assumption that genes are independent. We investigate the impact by simulating our empirical null distribution with correlation among
Figure 5.4: Examples of Correlations of two genes related to a GO terms from an experimental data set. $x$ axis and $y$ axis are the expression levels for the two genes for a particular sample. $r$ is the correlation between the genes over all. $r_n$ is the correlation between the genes for normal samples and $r_d$ is the correlation between genes for diseased samples. The red dots correspond to the diseased samples and the green dots correspond to the non-diseased samples.

genes and comparing the 95th percentile and 99th percentile cutoff with the theoretically derived null distribution. We generate our empirical data similar to what we did in Section 5.3.3 but with pairwise correlation, $\rho$. We select $\rho = 0.5$ to make sure that we have a significant correlation for sample size of 100 with type I error of $\alpha = 0.05$ and power of $\beta = 0.85$ (See Appendix F). We compare the tail percentile of the empirical null distribution with the theoretically approximated null distribution. The results are presented in Figure 5.5
and Figure 5.6. Ideally, if the correlation has no impact on the test statistics, the empirical null distribution simulated with gene by gene correlation should be close to the approximated null distribution without correlation. However, it is clearly not the case. When there is correlation between genes, the 99th and 95th percentile of the approximated null distribution is underestimating those of the empirical null distribution. This underestimation will make our test too liberal which will result in a higher type I error rate than what is controlled for in our analysis.

Figure 5.5: 95th percentile of $T_{[1,-1]}$ using approximation by normal, skew-$t$ and Edgeworth series. The circles in the figure are the 95th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points. The genes are simulated with pair-wise correlation coefficient of $\rho = 0.5$. 

166
Figure 5.6: 99th percentile of $T_{[1,-1]}$ using approximation by normal, skew-t and Edgeworth series. The circles in the figure are the 99th percentile of the empirical distribution of $T_{[1,-1]}$ obtained from 20000 simulated data points. The genes are simulated with pair-wise correlation coefficient of $\rho = 0.5$.

We can illustrate the impact of the correlation between genes through a power study. We simulated empirical alternative distributions against the approximated null using skew-t similar to Section 4.4.2, then we find the empirical power of different alternatives with the increase of the mean difference of the genes with or without pairwise correlation. The results are presented in Table 5.4 with non-correlated genes and in Table 5.5 with correlated genes.

In the second column of Table 5.4 and Table 5.5 under $\mu = 0$, these are the powers of our test under empirical null, which is essentially the controlled
Table 5.4: Power calculation by changing mean difference $\mu$ of all genes related to each GO with response variable $y = [1, -1]$. $H_0 : \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a : \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20. There is not correlation between genes.

<table>
<thead>
<tr>
<th>$m$</th>
<th>$\mu$</th>
<th>0</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.053(0.002)</td>
<td>0.053(0.002)</td>
<td>0.095(0.002)</td>
<td>0.184(0.003)</td>
<td>0.313(0.004)</td>
<td>0.475(0.004)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.052(0.002)</td>
<td>0.059(0.002)</td>
<td>0.117(0.003)</td>
<td>0.255(0.004)</td>
<td>0.453(0.004)</td>
<td>0.681(0.004)</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.053(0.002)</td>
<td>0.060(0.002)</td>
<td>0.131(0.003)</td>
<td>0.299(0.004)</td>
<td>0.557(0.004)</td>
<td>0.799(0.004)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.055(0.002)</td>
<td>0.063(0.002)</td>
<td>0.145(0.003)</td>
<td>0.352(0.004)</td>
<td>0.642(0.004)</td>
<td>0.875(0.003)</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.052(0.002)</td>
<td>0.061(0.002)</td>
<td>0.16(0.003)</td>
<td>0.402(0.004)</td>
<td>0.718(0.004)</td>
<td>0.927(0.002)</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.052(0.002)</td>
<td>0.062(0.002)</td>
<td>0.17(0.003)</td>
<td>0.436(0.004)</td>
<td>0.774(0.004)</td>
<td>0.955(0.002)</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.053(0.002)</td>
<td>0.063(0.002)</td>
<td>0.177(0.003)</td>
<td>0.478(0.004)</td>
<td>0.822(0.003)</td>
<td>0.974(0.001)</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.046(0.002)</td>
<td>0.059(0.002)</td>
<td>0.184(0.003)</td>
<td>0.513(0.004)</td>
<td>0.860(0.003)</td>
<td>0.985(0.001)</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.047(0.002)</td>
<td>0.060(0.002)</td>
<td>0.203(0.004)</td>
<td>0.552(0.004)</td>
<td>0.895(0.003)</td>
<td>0.992(0.000)</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.049(0.002)</td>
<td>0.063(0.002)</td>
<td>0.214(0.004)</td>
<td>0.595(0.004)</td>
<td>0.919(0.002)</td>
<td>0.995(0.000)</td>
<td></td>
</tr>
</tbody>
</table>

The type I error. The type I error should be close to 0.05 if the approximated null distribution is correct. In our results, the type I error is correct when there is no correlation between genes but incorrect when there is correlation. It is because the null distribution is derived under the independence assumption. As we stated earlier, when there is correlation added to the data, the theoretically derived null distribution is underestimating the true null distribution, resulting in increased type I error. In the power table, it shows exactly what is expected. It seems that the test procedure has better power when there is correlation in the data, but in fact these are the results of an inflated type I error. Our
Table 5.5: Power calculation by changing mean difference $\mu$ of all genes related to each GO with response variable $y = [1, -1]$. $H_0 : \mu_{0j} = \mu_{1j}, j = 1, \ldots, m$ versus $H_a : \mu_{0j} - \mu_{1j} = 2\mu, j = 1, \ldots, m$. The number of genes within each GO term is $m$. The numbers in the parentheses are the standard errors of the powers. The sample size used to generated the power is 20. There is correlation between genes, pairwise correlation $\rho = 0.5$

<table>
<thead>
<tr>
<th>$m^\lambda\mu$</th>
<th>0</th>
<th>0.1</th>
<th>0.3</th>
<th>0.5</th>
<th>0.7</th>
<th>0.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.053(0.002)</td>
<td>0.053(0.002)</td>
<td>0.095(0.002)</td>
<td>0.184(0.003)</td>
<td>0.313(0.004)</td>
<td>0.475(0.004)</td>
</tr>
<tr>
<td>2</td>
<td>0.061(0.002)</td>
<td>0.069(0.002)</td>
<td>0.128(0.003)</td>
<td>0.253(0.004)</td>
<td>0.426(0.004)</td>
<td>0.622(0.004)</td>
</tr>
<tr>
<td>3</td>
<td>0.064(0.002)</td>
<td>0.072(0.002)</td>
<td>0.150(0.003)</td>
<td>0.292(0.004)</td>
<td>0.494(0.004)</td>
<td>0.690(0.004)</td>
</tr>
<tr>
<td>4</td>
<td>0.080(0.002)</td>
<td>0.088(0.002)</td>
<td>0.167(0.003)</td>
<td>0.328(0.004)</td>
<td>0.529(0.004)</td>
<td>0.735(0.004)</td>
</tr>
<tr>
<td>5</td>
<td>0.083(0.002)</td>
<td>0.098(0.002)</td>
<td>0.189(0.003)</td>
<td>0.361(0.004)</td>
<td>0.572(0.004)</td>
<td>0.771(0.004)</td>
</tr>
<tr>
<td>6</td>
<td>0.093(0.002)</td>
<td>0.104(0.003)</td>
<td>0.196(0.003)</td>
<td>0.370(0.004)</td>
<td>0.595(0.004)</td>
<td>0.787(0.004)</td>
</tr>
<tr>
<td>7</td>
<td>0.096(0.002)</td>
<td>0.106(0.003)</td>
<td>0.211(0.004)</td>
<td>0.406(0.004)</td>
<td>0.623(0.004)</td>
<td>0.807(0.003)</td>
</tr>
<tr>
<td>8</td>
<td>0.101(0.003)</td>
<td>0.115(0.003)</td>
<td>0.221(0.004)</td>
<td>0.413(0.004)</td>
<td>0.640(0.004)</td>
<td>0.827(0.003)</td>
</tr>
<tr>
<td>9</td>
<td>0.104(0.003)</td>
<td>0.114(0.003)</td>
<td>0.231(0.004)</td>
<td>0.435(0.004)</td>
<td>0.663(0.004)</td>
<td>0.840(0.003)</td>
</tr>
<tr>
<td>10</td>
<td>0.102(0.003)</td>
<td>0.119(0.003)</td>
<td>0.233(0.004)</td>
<td>0.441(0.004)</td>
<td>0.667(0.004)</td>
<td>0.847(0.003)</td>
</tr>
</tbody>
</table>

approximated null distribution is incorrect for data in which there is correlation involved!
Chapter 6

Summary and Further Research

In our research, we start with trying to use the existing domain knowledge, i.e. the Gene Ontology to increase the interpretability of microarray data analysis. DEA-PLS is proposed in Section 3. It uses partial least square procedure to find summary measurement for the genes related to each GO term, and tests for the significance of each GO term by a two-sample t test on each summary measurement. According to simulation and experiment data analysis, this procedure can successfully detect significant GO terms. However, the two-sample t test is inadequate to adjust our test for different size of GO terms. A new testing procedure is proposed in Section 4. The new test relies on theoretically derived null distribution of the test statistics we proposed based on DEA-PLS. It not only provides us a valid testing procedure and a way to adjust the tests
according to different size of the GO term, but also simplifies the computation intensity. We then study the impact of different of codings of the response variables on our test procedure in Section 5. Since we have focused on data with binary response, the different codings are \( y = [1, -1] \) or \( y = [0, 1] \). We are able to find the equivalence of tests with the two type of coding. Along the way, we also extend the testing procedure to more generalized cases.

However, our testing procedure still operates under the assumption of independence between genes. As we have pointed out, this assumption is invalid in most cases. Some researchers have proposed functional analysis to incorporate the correlation structures among genes. Most recently, Delongchamp et al. (2006) proposed a Meta-analysis method to compute the overall statistical significance of a group of genes. Their method combines p-values of each gene into an overall significance level, which is adjusted for correlations among the genes. For our testing procedure, we can possibly find an adjusted test statistic that incorporates correlation. But this is not trivial because it involves estimating the correlation structures of large numbers of genes and representing it under the framework of our testing procedure. This is a very interesting topic and likely to lead to a test more practical for experimental data.
References


Arfin, S., Long, A., Ito, E., Toller, L., Richle, M., Paegle, E., and Hatfield,


Dahlquist, K. D., Salomonis, N., Vranizan, K., Lawlor, S., and Conklin, B.


Lee, M. T., Gray, R., Bjorkbacka, H., and Freeman, M. (2005b). General-
ized rank tests for replicated microarray data. *Statistical Applications in Genetics and Molecular Biology, 4*(1).


Tian, L., Greenberg, S., Kong, S., Altschuler, J., Kohane, I., and Park, P. (2005). Discovering statistically significant pathways in expression profil-


Welch, B. L. (1947). The generalization of `students’ problem when several different population variances are involved. *Biometrika*, **34**, 28–35.


Bioinformatics, 19, 1046–1054.

Appendices
Appendix A

Sample Codes for Running PLS Procedure in SAS

This is a piece of SAS code that runs PLS procedure with cross-validation on each of 990 gene sets and then output the meaningful first scores in an aggregated data set.

```sas
%macro datagen;
%do i=1 %to 990;
  data data&i;
    set mu03.simudata;
    if ( ref_id=&i );
  run;
  proc sort data=data&i nodupkey; by affy; run;
  proc transpose data=data&i out=data&i;
    var col1-col100;
  run;
  data temp&i ( drop=_name_ );
    set data&i;
    extra=0;
  run;
  data temp&i;
  merge temp&i mu03.Y;
  run;
/* crossvalidation*/
  proc pls data=temp&i cv=split cvtest(seed=12345) METHOD=SIMPLS noprin;
  model x=col1-extra; output out=outcv&i xscore=T;
  run;
  data outcv&i(keep=M&i);
```
set outcv&i;
MFr&i=T1;
run;
%end;
data mu03.simuStepcv&j;
set outcv1;
run;
%do i=2 %to 990;
data mu03.simuStepcv&j;
merge mu03.simuStepcv&j outcv&i;
run;
%end;
%end;
%mend datagen;
%datagen;
Appendix B

R code for Simulation Studies

Here is the R code for generating out simulation data sets for cases $\delta = 0.3$

```r
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
MF139.affy <- c(1514:1522)
MF384.affy <- c(2002)
MF415.affy <- c(1283)
MF601.affy <- c(2872,2874,2887,2888,2889)
important.affy <-c(MF139.affy, MF384.affy, MF415.affy, MF601.affy)
# Association between MF and affy
MF139 <- c(29)
MF384 <- c(215)
MF415 <- c(234)
MF601 <- c(335)
important.MF <-c(MF139, MF384, MF415, MF601)
MF142 <- c(33)
MF622 <- c(347)
MF725 <- c(399)
MF763 <- c(419)
MF931 <- c(486)
important.MF2 <-c(MF142, MF622, MF725, MF763, MF931)
p.sim <- 556
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
# # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # # #
# Importance of MF
my.delta <- 0.3
# | mean at y=0 - mean at y=1| = 2*my.delta
mean.MF139 <- my.delta
mean.MF384 <- 1.2*my.delta
mean.MF415 <- 1.8*my.delta
```

190
mean.MF601 <- 1.3*my.delta

# Simulate data
set.seed(123456)
n.sim <- 100; p.sim <- 3666
# y.sim <- rnorm(n=n.sim, mean=0, sd=1)
# y.sim.bin <- 1*(y.sim>=0)
y.sim.bin <- rep(0, 100)
y.sim.bin[51:100] <- rep(1, 50)
x.sim <- matrix(rnorm(n=n.sim*p.sim, mean=0, sd=1), nrow=n.sim, ncol=p.sim)
# n0 <- sum(y.sim.bin==0)
# n1 <- n.sim-n0
n0 <- 50
n1 <- 50

x.sim[y.sim.bin==0, MF139.affy] <- rnorm(n=n0*length(MF139.affy), mean=mean.MF139, sd=1)
x.sim[y.sim.bin==1, MF139.affy] <- rnorm(n=n1*length(MF139.affy), mean=-mean.MF139, sd=1)

x.sim[y.sim.bin==0, MF384.affy] <- rnorm(n=n0*length(MF384.affy), mean=mean.MF384, sd=1)
x.sim[y.sim.bin==1, MF384.affy] <- rnorm(n=n1*length(MF384.affy), mean=-mean.MF384, sd=1)

x.sim[y.sim.bin==0, MF415.affy] <- rnorm(n=n0*length(MF415.affy), mean=mean.MF415, sd=1)
x.sim[y.sim.bin==1, MF415.affy] <- rnorm(n=n1*length(MF415.affy), mean=-mean.MF415, sd=1)

x.sim[y.sim.bin==0, MF601.affy] <- rnorm(n=n0*length(MF601.affy), mean=mean.MF601, sd=1)
x.sim[y.sim.bin==1, MF601.affy] <- rnorm(n=n1*length(MF601.affy), mean=-mean.MF601, sd=1)

x.full <- x.sim
for (i in 1:49) {
  x.sim <- matrix(rnorm(n=n.sim*p.sim, mean=0, sd=1), nrow=n.sim, ncol=p.sim)
  n0 <- sum(y.sim.bin==0)
  n1 <- n.sim-n0
	x.sim[y.sim.bin==0, MF139.affy] <- rnorm(n=n0*length(MF139.affy), mean=mean.MF139, sd=1)
x.sim[y.sim.bin==1, MF139.affy] <- rnorm(n=n1*length(MF139.affy), mean=-mean.MF139, sd=1)
  x.sim[y.sim.bin==0, MF384.affy] <- rnorm(n=n0*length(MF384.affy), mean=mean.MF384, sd=1)
x.sim[y.sim.bin==1, MF384.affy] <- rnorm(n=n1*length(MF384.affy), mean=-mean.MF384, sd=1)
  x.sim[y.sim.bin==0, MF415.affy] <- rnorm(n=n0*length(MF415.affy), mean=mean.MF415, sd=1)
x.sim[y.sim.bin==1, MF415.affy] <- rnorm(n=n1*length(MF415.affy), mean=-mean.MF415, sd=1)
  x.sim[y.sim.bin==0, MF601.affy] <- rnorm(n=n0*length(MF601.affy), mean=mean.MF601, sd=1)
x.sim[y.sim.bin==1, MF601.affy] <- rnorm(n=n1*length(MF601.affy), mean=-mean.MF601, sd=1)
}
x.sim[y.sim.bin==0, MF384.affy] <- rnorm(n=n0*length(MF384.affy), mean=mean.MF384, sd=1)
x.sim[y.sim.bin==1, MF384.affy] <- rnorm(n=n1*length(MF384.affy), mean=-mean.MF384, sd=1)

x.sim[y.sim.bin==0, MF415.affy] <- rnorm(n=n0*length(MF415.affy), mean=mean.MF415, sd=1)
x.sim[y.sim.bin==1, MF415.affy] <- rnorm(n=n1*length(MF415.affy), mean=-mean.MF415, sd=1)

x.sim[y.sim.bin==0, MF601.affy] <- rnorm(n=n0*length(MF601.affy), mean=mean.MF601, sd=1)
x.sim[y.sim.bin==1, MF601.affy] <- rnorm(n=n1*length(MF601.affy), mean=-mean.MF601, sd=1)

x.full <- rbind(x.full, x.sim)
Appendix C

R codes for Generating Mapping from Genes to GO

Here is the R code to generate mappings from genes to GO terms using the bio-conductor package.

```
# library (Biobase)
library (tools)
library (ALL)
library (GO)
library (annotate)
library (xtable)
library (genefilter)
library (RBGL)
library (Ruuid)
library (graph)
library (GOstats)
library (limma)

xx <- as.list(hgu95av2GO) test <- as.data.frame(unlist(xx)) write(test, file = 'E:\\users\\jj\\temp.txt', sep = '\t')

### get affy to GO mapping from R, incomplete version (without propagation)########
# Convert to a list
xx <- as.list(hgu95av2GO)
# Remove all the NAs
xx <- xx[!is.na(xx)]
if(length(xx) > 0){
```

193
Try the first one

got <- xx[[1]]
got[[1]][['GOID']]
got[[1]][['Ontology']]
got[[1]][['Evidence']]

} affylist <- names(xx) affy_N <- length(xx) xx <- as.list(hgu95av2GO)

# Remove all the NAs
xx <- xx[!is.na(xx)]

i=1

got <- xx[[i]]
go_N <- length(got)
mapping <- matrix(0, go_N, 3)
for (j in 1:go_N) {
  mapping[j,1] <- names(xx)[i]
  mapping[j,2] <- got[[j]][['GOID']]
  mapping[j,3] <- got[[j]][['Ontology']]
}

for (i in 2:affy_N) {
  got <- xx[[i]]
go_N <- length(got)
temp <- matrix(0, go_N, 3)
for (j in 1:go_N) {
  temp[j,1] <- names(xx)[i]
  temp[j,2] <- got[[j]][['GOID']]
  temp[j,3] <- got[[j]][['Ontology']]
}
  mapping <- rbind(mapping, temp)
}

write.table(as.data.frame(mapping), file =
'E:\users\jj\markerdata\bioconductor\R_mapping.txt' ,
194
### get affy to GO mapping from R, complete version (propagation)

```r
xx <- as.list(hgu95av2GO2ALLPROBES)
xx <- xx[!is.na(xx)]
GOlist <- names(xx) N <- length(xx)

i = 1
n <- length(xx[[1]])
mapping <- matrix(0, n, 2)
for (j in 1:n)
    mapping[j, 1] <- GOlist[i]
mapping[j, 2] <- xx[[i]][j]

for (i in 2:N)
    n <- length(xx[[i]])
    temp <- matrix(0, n, 2)
    for (j in 1:n)
        temp[j, 1] <- GOlist[i]
        temp[j, 2] <- xx[[i]][j]
    mapping <- rbind(mapping, temp)

write.table(as.data.frame(mapping), file =
'E:\users\jj\markerdata\bioconductor\R_mapping_complete.txt',
sep = '\t')
```

### get affy to locusID mapping

```r
xx <- as.list(hgu95av2LOCUSID)
# Remove probe ids that do not map to any LOCUSID
xx <- xx[!is.na(xx)]
```
fram <- as.data.frame(xx)
mat <- data.matrix(fram)
mat[1, 1:12238]

temp <- cbind(colnames(fram), mat[1, 1:12238])
Appendix D

R Codes for Simulating Empirical Null Distributions

Here is the R code for generating the empirical null distribution of the test statistics in Section 4.4.1.

```r
# Critical
critical <- matrix(0, 60, 4)

plot(0, 0, xlim=c(-10, 10), ylim=c(0, 1.2))
for (m in 1:60)
{
  MC <- 20000
  n <- 10
  j <- 2
  mu <- 0
  mu1 <- mu
  mu2 <- -mu
  set.seed(1234567)
  u1 <- rep(mu1, n)
  u2 <- rep(mu2, n)
  # Simulating data under null#
  X1 <- matrix(c(rnorm(n=n*m*MC, mean=mu1, sd =1), rnorm(n=n*m*MC, mean=mu2, sd =1)), nrow=2*n*m, ncol=MC, byrow=TRUE)
  # Creating matrix A####
  temp <- diag(m)
  for (i in 1:(n-1))
  {
    temp <- cbind(temp, diag(m))
  }
  t1 <- temp
  for (i in 1:(n-1))
  {  
```
t1<-rbind(t1, temp)

}
A<-rbind(t1,-t1)
A<-cbind(matrix(0,2*n*m,n*m), A)
T<-rep(0,MC)

## creating null distribution for test statistics ##
for (i in 1:MC){
  T[i]<-(t(X1[,i])%*%A %*% X1[,i]/10-(-1*m))/(3*m)^0.5)
} critical [m,1]<-mean(T) critical [m,2]<-var(T)
critical [m,3]<-sort(T)[MC*0.05] critical [m,4]<-qnorm(0.05, mean=0, sd=1, log.p = FALSE)
write.table(critical, file = 'C:\\users\\jj\\tex\\IBI\\extrawriteup\\critical.csv', sep=' ', col.names = NA)
c<-read.table('C:\\users\\jj\\tex\\IBI\\extrawriteup\\critical.csv', sep=' ', header=T)
plot(1:60,c[,5], type='l', ylim=c(-2.2,-1.6)) points(1:60,c[,4])

# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Appendix E

Unbalanced Case with Mean Difference $\mu_0$ and $\mu_1$ and variance $\sigma_j$

In the case that we have unbalanced sample sizes, $n_0$ for $y = -1$ and $n_1$ for $y = 1$, we assume the $j$th gene expression $w_{..j} \sim iid \sim N(\beta_j, \sigma_j^2 I)$, \[ \beta_j = \begin{bmatrix} \mu_{0j} 1_{n_0} \\ \mu_{1j} 1_{n_1} \end{bmatrix}, j = 1, \ldots, m. \]

We center and scale our expression level by subtracting it by its mean and dividing by the pooled sample standard error, \[ x_{..j} = \frac{w_{..j} - \bar{w}_{..j} 1_{n_0+n_1}}{\hat{\sigma}_j} = (I - P)w_{..j} \frac{1}{\hat{\sigma}_j}, \]

where $P = \frac{1}{n_0+n_1} 1_{n_0+n_1} 1_{n_0+n_1}^T$ is a $(n_0 + n_1) \times (n_0 + n_1)$ projection matrix and hence $I - P$ is also a projection matrix. $\hat{\sigma}$ is defined similarly as for unbalanced classes.

199
Therefore,

\[
\overline{X}_{0,j} = \frac{1}{n_0} \left[ \begin{array}{c} 1_{n_0}^T \\ 0_{n_1}^T \end{array} \right] x_{-j} \\
= \frac{1}{n_0} \left[ \begin{array}{c} 1_{n_0}^T \\ 0_{n_1}^T \end{array} \right] (I - P) w_{-j} \frac{1}{\sigma_j} \\
= \frac{1}{n_0} \left\{ \left[ \begin{array}{c} 1_{n_0}^T \\ 0_{n_1}^T \end{array} \right] - \left[ \begin{array}{c} 1_{n_0}^T \\ 0_{n_1}^T \end{array} \right] \frac{1}{n_0 + n_1} 1_{n_0+n_1} \right\} w_{-j} \frac{1}{\sigma_j} \\
= \frac{1}{n_0} \left\{ \left[ \begin{array}{c} 1_{n_0}^T \\ 0_{n_1}^T \end{array} \right] - \frac{n_0}{n_0 + n_1} 1_{n_0+n_1} \right\} w_{-j} \frac{1}{\sigma_j} \\
= \frac{1}{n_0} \left[ \frac{n_1}{n_0+n_1} 1_{n_0}^T - \frac{n_1}{n_0+n_1} 1_{n_1}^T \right] w_{-j} \frac{1}{\sigma_j} \\
= \frac{1}{n_0} \left( \frac{n_1}{n_0+n_1} \overline{W}_{0,j} - \frac{n_0}{n_0+n_1} n_1 \overline{W}_{1,j} \right) \frac{1}{\sigma_j} \\
= \frac{n_1}{n_0+n_1} (\overline{W}_{0,j} - \overline{W}_{1,j}) \frac{1}{\sigma_j},
\]

Similarly,

\[
\overline{X}_{1,j} = -\frac{n_0}{n_0+n_1} (\overline{W}_{0,j} - \overline{W}_{1,j}) \frac{1}{\sigma_j}.
\]
To test for $H_0 : \mu_0 = \mu_1$, we define our test statistics as

$$T_{[1,-1]} = \left[ -\mathbf{1}_{n_0}^T \mathbf{1}_{n_1}^T \right] S \| X^Ty \|
$$

$$= \left( \frac{n_0 + n_1}{2n_0n_1} \right)^2 \left( \left[ -\mathbf{1}_{n_0}^T \mathbf{1}_{n_1}^T \right] X \right) \left( X \left[ \begin{array}{c} -\mathbf{1}_{n_0}^T \\ \mathbf{1}_{n_1}^T \end{array} \right] \right)
$$

$$= \sum_{j=1}^m (n_0 \bar{X}_{0,j} - n_1 \bar{X}_{1,j})^2
$$

$$= \sum_{j=1}^m \left( n_0 - \frac{n_1}{n_0 + n_1} \left( \bar{W}_{0,j} - \bar{W}_{1,j} \right) \frac{1}{\hat{\sigma}_j} + n_1 - \frac{n_0}{n_0 + n_1} \left( \bar{W}_{0,j} - \bar{W}_{1,j} \right) \frac{1}{\hat{\sigma}_j} \right)^2
$$

$$= \sum_{j=1}^m \left( \frac{2n_0n_1}{n_0 + n_1} \left( \bar{W}_{0,j} - \bar{W}_{1,j} \right) \frac{1}{\hat{\sigma}_j} \right)^2
$$

$$= \left( \frac{2n_0n_1}{n_0 + n_1} \right)^2 \sum_{j=1}^m \left( \bar{W}_{0,j} - \bar{W}_{1,j} \right)^2
$$

Conveniently, our new $T_{[1,-1]}$ has the similar form as 5.11 with a sum of $F$ distribution multiplied by some constants. Note the test statistic in 5.11 is just a special case of the test statistic for unbalanced case.
Appendix F

Derivation of Significant Correlation Coefficients

To ensure we have significant pairwise correlations for gene expressions with sample size 100 and $\alpha = 0.05$ and power of $\beta = 0.85$, we derive the pairwise correlation here.

The correlation test is constructed as:

$$T = 0.5 \log \frac{1 + r}{1 - r}.$$ 

Under null, this test statistics is distributed as a normal distribution with mean $\mu = 0.5 \log \frac{1 + \rho}{1 - \rho}$, standard deviation $\sigma = \frac{1}{\sqrt{n-3}}$, where $r$ is the correlation, $\rho$ is the correlation coefficient and $n$ is the sample size. To ensure our test have an
type I error of $\alpha$ and power of $\beta$, we have

\[
\Pr \left( T > \frac{Z_\alpha \sigma}{\sqrt{n}} \mid H_0 \text{is true} \right) \geq 1 - \beta
\]

\[
\Pr \left( \frac{T - 0.5 \log \frac{1+\rho}{1-\rho}}{\sqrt{n-3}} > \frac{Z_\alpha \frac{1}{\sqrt{n}} - 0.5 \log \frac{1+\rho}{1-\rho}}{\sqrt{n-3}} \right) \geq 1 - \beta
\]

\[
\Pr \left( Z > \frac{Z_\alpha}{\sqrt{n}} - 0.5 \sqrt{n-3} \log \frac{1+\rho}{1-\rho} \right) \geq 1 - \beta
\]

\[
1 - \Pr \left( Z \leq \frac{Z_\alpha}{\sqrt{n}} - 0.5 \sqrt{n-3} \log \frac{1+\rho}{1-\rho} \right) \geq 1 - \beta
\]

\[
1 - \Phi \left( Z \leq \frac{Z_\alpha}{\sqrt{n}} - 0.5 \sqrt{n-3} \log \frac{1+\rho}{1-\rho} \right) \geq 1 - \beta
\]

\[
\Phi \left( Z \leq \frac{Z_\alpha}{\sqrt{n}} - 0.5 \sqrt{n-3} \log \frac{1+\rho}{1-\rho} \right) \leq \beta
\]

\[
\frac{Z_\alpha}{\sqrt{n}} - 0.5 \sqrt{n-3} \log \frac{1+\rho}{1-\rho} \leq -Z_\beta.
\]

We get

\[
\rho \geq \frac{\exp \left( \frac{2(Z_\alpha + Z_\beta)}{\sqrt{n-3}} \right) - 1}{\exp \left( \frac{2(Z_\alpha + Z_\beta)}{\sqrt{n-3}} \right) + 1}
\]

Therefore to have significant correlation coefficient with type I error of $\alpha = 0.05$ and power $\beta = 0.85$, where $Z_{0.025} = Z_{0.025} = 1.96$ and $Z_{0.05} = 0.85$, we need to have $\rho \geq 0.1058$. In our simulation we choose $\rho = 0.5$ as the pairwise correlation coefficient.