uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Li Mei Sun

AUTEUR DE LA THÈSE / AUTHOR OF THESIS

Ph.D. (Mathematics)

GRADE / DEGREE

Department of Mathematics and Statistics

FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

Experimental Designs for Microarray Experiments

TITRE DE LA THÈSE / TITLE OF THESIS

Gail Ivanoff

DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

Mayer Alvo

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Mohamedou Ould-Haye

Mary Thompson

Gary W. Slater

Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

EXPERIMENTAL DESIGNS FOR MICROARRAY EXPERIMENTS

By
Limei Sun, B.Sc., M.Sc.
April 2008

A Thesis
submitted to the Faculty of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy in Mathematics[1]

[1]The Ph.D. Program is a joint program with Carleton University, administered by the Ottawa-Carleton Institute of Mathematics and Statistics

Your file Votre référence
ISBN: 978-0-494-41647-1
Our file Notre référence
ISBN: 978-0-494-41647-1

Abstract

When more than two treatments are compared for each gene in microarray experiments, a good design should allocate the treatments in such a way as to collect more information on treatment comparisons of interest. We compared a balanced incomplete block design, a loop design, and a reference design in terms of power to detect a single differentially expressed gene, where each design uses the same number of slides to compare the same number of treatments. A simple 2-way model is applied. Using a standard F test statistic for each single-gene analysis, the efficiency of each design was related to the size of the non-centrality parameter of the F distribution under a specified alternative. In preparation for the simultaneous analysis of multiple genes, the exact density function of the p-values under the alternative hypothesis was computed.

Microarray experiments consider thousands of genes simultaneously and thus the multiple testing issue arises. We can compute individual p-values for each gene, and both the False Discovery Rate (FDR) method and the Westfall-Young resampling method are multiple testing procedures based on ordered p-values. For the FDR method, we derive lower bounds for the probability of rejecting more than k genes, and for the probability that more than j out of m_1 truly differentially expressed genes are rejected using the FDR method. We simplify the expression of the lower bound under certain conditions by applying results from the weak convergence of extremal processes: the number of p-values falling into a small neighborhood of 0 has an asymptotic Poisson distribution. These probability estimates can be used to estimate the number of slides needed in experiments. The Westfall-Young method adjusts the observed p-values which are then compared with a pre-determined α. Under the

independence assumption, the Westfall-Young method is equivalent to the step-down Sidak method. It involves time consuming simulations to use the step-down Sidak method in experimental designs. We apply results from the weak convergence of extremal processes and propose a way to speed up the simulation for the first few k adjusted p-values (where k is a small number relative to the total number of p-values). Their asymptotic distributions are evaluated analytically. An alternative method to adjust p-values suggested by an anonymous reader is also discussed.

We numerically illustrate the applications of the procedures discussed in this thesis on a small data set from a spotted microarray experiment. These results are also illustrated in planning a non-spotted Affymetrix array experiment where they help determine the sample size needed to achieve a given level of performance in the FDR or Westfall-Young approaches.

Acknowledgements

I am deeply indebted to my supervisor, Dr. Andre Dabrowski, without whose encouragement, advice, assistance and financial support this thesis could not have been completed. He has been very generous with his ideas and time. I also thank him for his very kind concern about my life and career in these years.

I want to thank sincerely Dr. Gail Ivanoff for her understanding, help, and support. I also thank her for her care about my life. I also want to thank sincerely Dr. Mayer Alvo for his encouragement, help and support. Drs. Gail Ivanoff and Mayer Alvo provided all the help and support I needed after Dr. Andre Dabrowski passed away. Without their help, my thesis could not have been finished.

I am grateful to Drs. M.Zarepour and M.O.Haye, the other members of my supervisory committee. I am grateful to the Natural Sciences and Engineering Research Council, the School of Graduate Studies, and the department of Mathematics and Statistics for financial support in the form of Graduate Scholarship and Teaching Assistantships.

Dedication

Dedicated to my supervisor, Dr. Andre Robert Dabrowski, who unfortunately passed away in October 2006 before the defence of my thesis.

Contents

Chapter 1

Introduction

In this thesis, we focus on microarray experiments and study several properties of multiple testing procedures.

Chapter 2 provides an introduction to the biological background for microarray data, experimental designs, and statistical methods used on microarray data. The novel statistical feature of microarray experiments is that we can expect observations on ten to thirty thousand separate genes for each subject.

In Chapter 3 we compare three different designs commonly used for microarray experiments from the point of view of a single gene. Current cDNA spotted microarray slides have a two-dye system and each slide can be used to measure the relative abundance of genes from only two treatment groups. When more than two conditions (treatments) are being compared for each gene, the issue arises of how to assign these treatments on the slides. We consider 2-way linear models on the microarray data for each single gene and compare the power to detect a single differentially expressed gene using three different but commonly-used designs - the BIB, loop, and reference designs. In testing the same alternative hypotheses with the same number of slides, the different designs we consider have F test statistics with different non-centrality parameters. Consequently, in Chapter 4 the power of each design to detect a differentially expressed gene can be compared with that of the others. The exact density functions of p-values of the F statistics under the alternative hypothesis are developed.

In microarray experiments, thousands of genes are tested simultaneously and multiple testing issues arise. Some multiple testing procedures, such as the FDR method or the Westfall-Young resampling method, have been applied to control the false discovery or the family wise error rate. These two methods are both based on ordered p-values. In general these methods are conservative and the chance of detecting differentially expressed genes is small. Further, the cost of microarry slides is such that microarray experiments have tended to be rather limited in terms of the number of subjects. Consequently, many of these experiments have been disappointing in terms of their capacity to discover underlying associations between genes and conditions.

Therefore, it will be helpful in the planning stage of the experiment to know the anticipated probability of rejecting a certain number of hypotheses, say k, using these multiple testing methods. In Chapter 4 we derive a lower bound on the probability of rejecting k genes for the FDR method, under some assumptions. We also derive a lower bound for the probability that more than j rejections are true rejections (i.e. rejected genes are truly differentially expressed) given that a fixed number of hypotheses are false. These lower bounds could be used to estimate the number of required slides for a specified number of rejections or for a specified number of true rejections. One needs to determine the test statistic for each single gene to be used and to develop the distribution of p-values under the alternative hypothesis in order to calculate the lower bound.

The Westfall-Young resampling method is popular but has the disadvantage of being very computationally intensive. Under the assumption that all single gene tests are independent, it becomes the step-down Sidak method. To apply the step-down Sidak method in experimental designs, one needs to simulate all p-values a large number of times, which is time consuming. In Chapter 5, we apply results from the weak convergence of extremal processes and propose a way to quickly approximate the first k adjusted p-values by simulating only k exponential variables. Here k is a small number relative to the total number of p-values and usually these are the p-values of interest. The asymptotic distributions of the first few adjusted p-values can be evaluated analytically. These distributions could be used to estimate the sample size needed at the design stage. An alternative adjustment suggested by an anonymous

reader is discussed. It is less conservative but could lead to a large number of false rejections.

The results of each chapter are illustrated on a small data set of three spotted slides. In Chapter 6 we revisit that running example and provide a more complete discussion. We also consider planning a future experiment using non-spotted Affymetrix arrays and show how the results of this thesis can be employed to specify the number of subjects needed to achieve certain levels of performance of the multiple testing procedures.

The results of weak convergence of extremal processes are applied on the first k ordered p-values and listed in Appendix A. Under certain assumptions, when the number of tests is large enough (which is reasonable to assume for microarray experiments), the number of p-values falling in a small neighborhood of 0 has an asymptotic Poisson distribution. These distributional results can be applied on the asymptotic performance of multiple testing procedures and simplify the expressions or simulations used.

The contribution of this thesis comprises ways of evaluating the FDR method and the definition of criteria for use in planning experiments with high multiplicity of tests, ways of simulating the Westfall-Young method, and the implementation of such methods for realistic data sets.

Chapter 2

Introduction to Microarrays

Microarray technology has been developed recently to examine thousands of genes simultaneously. It detects which gene is expressed in a cell through measuring the amount of mRNA associated with that gene. In this chapter, we briefly review some biological background, experimental design, and statistical analysis of microarray data.

2.1 Biological Background

This section provides a brief review of the biology needed to understand the goal of microarray experiments. The material of this section was drawn from Haggis [14], Clote [6], Sebastiani et al [30], and Dorman [32]. The reader wishing greater detail can consult these references.

All living organisms are composed of one or more cells. Each cell is a complex system consisting of many different building blocks, such as a nucleus, smooth endoplasmic reticulum, rough endoplasmic reticulum, cytoplasm, cell membrane and so on. The nucleus of each cell contains all of the information the cell needs to perform specific functions, to grow and to divide. The chromosomes are the part of the nucleus that carries genes. Each gene is a functional and physical unit of heredity passed from parent to offspring. It occupies a specific locus on a chromosome. Genes are sequences of DNA and most genes contain the information for making a specific

protein and thus cells.

DNA is one kind of nucleic acids, which are polymers of nucleotides (Figure 1). Each nucleotide comprises a phosphate group, a sugar ring, and a base. Nucleotides are joined together by phosphate ester bonds between the phosphate of one nucleotide and the sugar of the next nucleotide. The sugar of a nucleotide is either ribose (for RNA) or deoxyribose (for DNA). They both contain 5 carbons and the carbons can be counted clockwise as C1', C2', C3', C4', and C5'. A nucleic acid has a free C3', which has a -OH group on it and the other a free C5' with a phosphate group on it. Therefore, nucleic acid always has a discernible orientation, which is conventionally depicted in the 5' to 3' direction. The phosphate groups and the sugars for all the DNAs (or RNAs) are the same and thus do not carry any information. They form the backbone of a nucleic acid and hold the bases in their proper places. On the other hand, nucleotides differ from each other because of the different bases they have and all the genetic information carried by a nucleic acid is encoded by the sequence of bases (or equivalently, the sequence of nucleotides). The bases found in naturally occurring nucleic acids are either purine or pyrimidine. Purines include adenine (A) and guanine (G), and pyrimidines include cytosine (C) and thymine (T) (in DNA) or uracil (U) (in RNA). Purines have a two-ring structure whereas pyrimidines have a single ring structure. A purine always pairs with a pyrimidine: A always pairs with T and G always pairs with C.

Nucleic acids are of two kinds: deoxyribonucleic acid (DNA) and ribonucleic acid (RNA). A DNA molecule (Figure 2) is a helical double-stranded chain of nucleotides with a hydrogen (-H, deoxyribonucleotides) group on each sugar ring. The sugar phosphate forms the backbone on the exterior and the bases are toward the interior. The bases guanine and cytosine are complementary and pair up using three hydrogen bonds, and the complementary pair of adenine and thymine only forms two hydrogen bonds. The two DNA strands are held together by hydrophobic interactions and hydrogen bonds between complementary base pairs. The strands are antiparallel in that they line up with one in the 5' to 3' direction and the other in the 3' to 5' direction. Two such strands are called complementary - that is, one can be obtained from the other by mutually exchanging A with T and C with G, and changing the

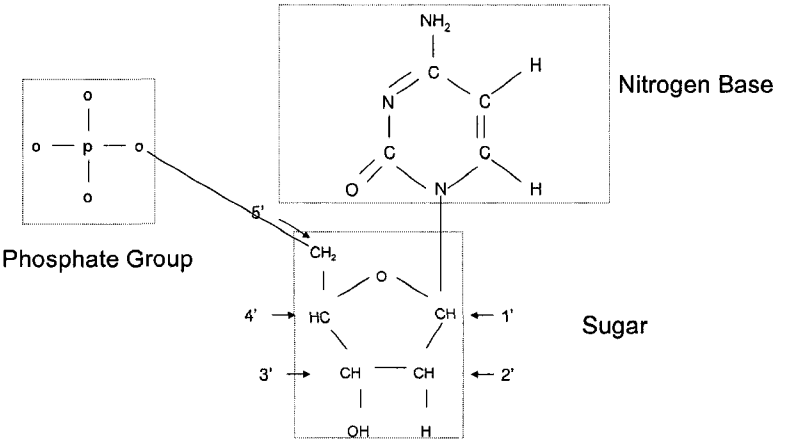

Figure 1: Nucleotides unit

direction of the molecule to the opposite. Since the DNA strands are complementary, each one of them fully determining the other, it is enough to give only one strand of the genome molecule for the purpose of information. Single-stranded DNA is the foundation for the microarray technology that will be described later.

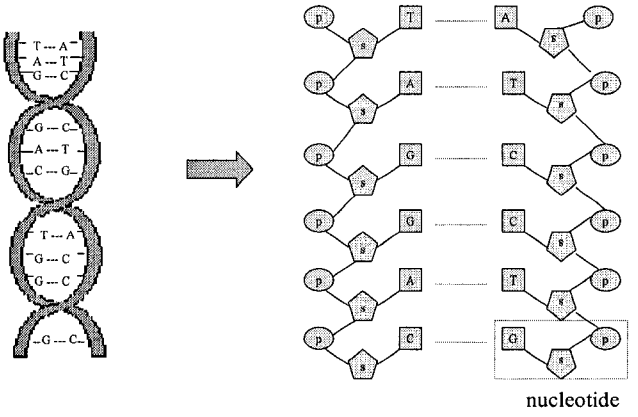

Figure 2: Nucleotide unit and DNA structure.

DNA is replicated before a cell can divide. Replication is the process by which an identical copy of DNA is made. Replication occurs every time a cell divides so that information can be preserved and handed down to offspring. DNA replication

begins with a partial unwinding of the double helix. An enzyme, DNA polymerase, binds to one strand of DNA and moves along it in the 3' to 5' direction. This DNA strand is used as a template, i.e. the DNA polymerase reads the sequence of bases on the template and then synthesizes the complementary strand and reforms a double helix. Another DNA polymerase binds to the other DNA strand as the double helix opens. However, it must grow in the opposite direction and is synthesized in small pieces as the original DNA stands are antiparallel and synthesis can only occur in the 5' to 3' direction. Enzyme DNA ligase joins the pieces of DNA and reforms another double helix. Replication proceeds until copies of each strand are made. When the replication process is complete, two identical DNA molecules that are also identical to the original DNA have been produced. DNA replication is semi-conservative: each copy contains one of the original strands paired with a newly synthesized strand that is complementary in terms of A-T and G-C base pairing.

DNA stores genetic information and passes it on to RNA. Like DNA, RNA is also constructed from nucleotides. However, the hydrogen (-H) group on each sugar ring is replaced by -OH, and the base T is replaced with pyrimidine Uracil (U). RNA is a single stranded chain. It can bind with a complementary single stranded DNA with U replacing T. The function of RNA is to read, decode and use the information received from DNA to make proteins. Typical types of RNA are messenger RNA (mRNA), transfer RNA (tRNA) and ribosomal RNA (rRNA). They differ in forms and structures. mRNA transcribes genetic information from DNA to a protein. tRNA carries amino acids corresponding to their codon (a sequence of three nucleotides bases that provides genetic code information for a particular amino acid) and is used in the translation of RNA to protein. Further, rRNA forms ribosomes for protein synthesis.

As we have mentioned, genes are working subunits of DNA and most genes contain the information for making proteins. Protein is synthesized through transcription and translation and is the key to making our bodies function. To make a protein from a gene, first the DNA (gene) strands are separated. One strand acts as the template and is copied into a mRNA according to the rule of complementary base pairs with U replacing T. This process is called RNA synthesis and through it the DNA is transcribed. After transcription, the mRNA leaves the cell nucleus and carries the

message to the site of translation - cellular cytoplasm.

Amino acid, which is the basic unit of a protein, is presented by a 3-nucleotide codon on the mRNA. Every tRNA has a specific 3-nucleotide sequence called anticodon and a site to which a specific amino acid binds. The 3-nucleotide codons on mRNA can be recognized by the anticodon of a tRNA according to the complementary base pairing rule (i.e. A=U and C=G). On the other hand, ribosomes, the cellular assemblies at which proteins are synthesized, are specialized in matching the anticodon of a tRNA to the corresponding codon on a mRNA and catalyzing the addition of the amino acids carried by individual tRNAs to the end of a growing peptide chain as well. As a result, proteins are synthesized at ribosomes according to the amino acid sequence encoded by the nucleotide sequence of an mRNA, which is copied from a DNA sequence via transcription.

In summary, the information carried in a gene (DNA sequence) is expressed by first transcribing into mRNA and then translating into proteins. This process is called gene expression (Figure 3). The gene expression is assumed to be controlled at various points in the sequence leading to protein synthesis and this control is thought to be the major determinant of cellular differentiation.

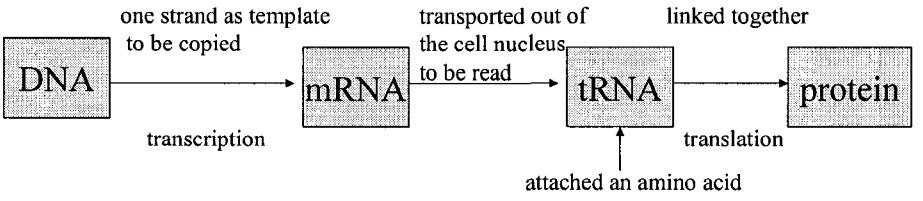

Figure 3: Protein synthesis.

To find out which genes are expressed in a particular cell type of an organism, at a particular time, under particular conditions, traditional biological methods focused on a "one gene in one experiment" basis. Although it remains the best approach to obtain knowledge of a single gene function, it only provides very limited and isolated information and the "whole picture" of gene function is hard to obtain. In the past several years, a new technology, called the DNA microarray, was developed to attach

the whole genome on a single slide so that researchers can have a better picture of the interactions among thousands of genes simultaneously. The method is still not perfect since not all genes are equally active, and the activity can be turned on or off in response to stimulation. However, it provides a way to look at thousands of genes in one experiment. The idea of a microarray is to detect which genes are expressed through measuring the amount of mRNA associated with each gene.

A spotted microarray (Figure 4) is typically a glass (or other material) slide, with thousands of DNA sequences attached on it as spots by a special robotic arm. Any given spot contains only one particular DNA strand sequence and each sequence may be spotted multiple times per array. Microarrays in current use measure from 1,000 to 25,000 genes. The sample spot sizes are typically less than 200 microns in diameter.

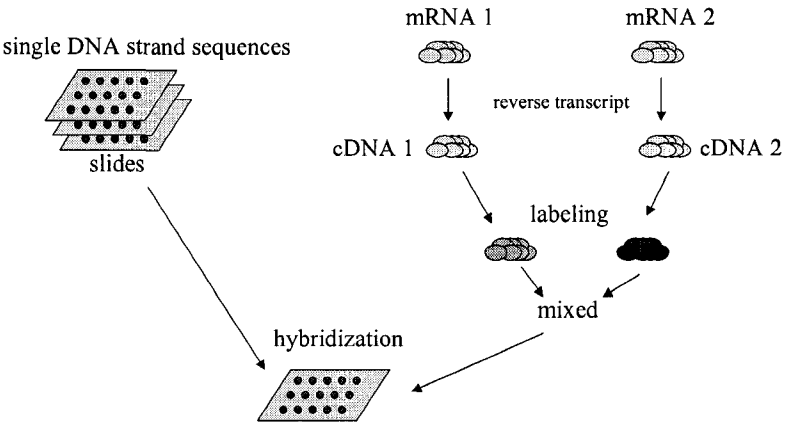

Figure 4: Microarray Experiment.

To compare the relative abundance of each gene sequence in two mRNA samples under study, the mRNA samples are first reverse-transcribed to a complementary DNA (cDNA) clones and labeled with two different fluorescent colours (usually a red dye and a green dye). They are then mixed and washed over the arrayed DNA spots. Since complementary single stranded nucleic acid sequences tend to attract each other, the dye-labeled cDNA strands can hybridize to their complementary sequences in the spots following the base-pairing rule. The fluorescence for each dye in each DNA spot is measured. The properties of genes can be explored when they have differential

expression at the mRNA level: the sample that contained more transcripts should produce the higher signal. If the RNA from the red sample is in abundance, the spot will be red. If the RNA from the green sample is in abundance, the spot will be green. If two RNA samples are equally abundant, the spot will appear yellow. If no samples bind, the spot will appear dark.

The DNA microarray technology provides a medium for matching known and unknown DNA samples based on base-pairing rules, and automates the process of identifying the unknowns. It is mainly applied to identify sequences of genes (for instance, to provide information for comparison of gene expression between normal and diseased cells), or to determine the expression level (RNA abundance) of genes.

The whole microarray experiment process can be summarized as follows: given the biological questions of interest, researchers design experiments, print the microarrays, and hybridize them. The images are processed, and expression data are extracted and then analyzed. The results will be used to make some biological conclusions that might lead to more biological questions. Usually statisticians can be involved in the experimental design and data analysis steps. We will focus on the experiment design step in this thesis.

Another major type of array is the Affymetrix array which uses Affymetrix technology. This technology incubates a series of deoxynucleotides on the surface of a glass one after another through photolithographic fabrication. Each series is a sequence of nucleotides (usually up to approximately 20 nucleotides) corresponding to a part of a gene under investigation. The synthesized nucleotides sequences are called probes and each sequence is represented by two probes: one with the exact sequence of the chosen fragment of the gene (perfect match or PM) and one with a mismatch nucleotide in the middle of the fragment (mismatch of MM). These two probes are always synthesized adjacent to each other to control for spatial differences for hybridization. The sample of interest is hybridized to the array at a critical temperature when the PM hybridizes and even a single mismatch prevents the MM from hybridizing. The signals of both probes are measured and the average differences between the PM and MM is commonly used as the value that represents the expression level of the gene. This material has been taken from Draghici [10].

These two types of arrays have their own advantages: Affymetrix arrays provide more reliable data, but only spot short known sequences due to the limitation of the technology. cDNA arrays are more flexible, allow spotting long sequences, and are easier to analyze with appropriate experimental designs. In this thesis we will focus on cDNA arrays.

2.2 Data Acquisition

Once the labeled mRNA is hybridized on the array, the array is scanned to measure the intensity of the spots for each dye. This is done in two steps. First, the array is illuminated with a laser light that excites one fluorescent (e.g. the red dye). An image is captured and theoretically, the intensity of each spot is proportional to the amount of the red mRNA with the sequence matching the given spot. Second, the array is illuminated again with a laser light, but this time the laser light has a frequency that excites the other fluorescent dye (e.g. green). Another image is captured and the intensity of each spot will be proportional to the amount of the green mRNA. These two images are black and white. They are colored to the corresponding color of the mRNA (red and green) by some software for visualization. Then these two images are overlaid to produce an image in which spots will have colors from green through yellow to red. See Figure 5, for example. The intensity of each gene on every spot for each color is measured and is used to represent the abundance of the gene in the mRNA sample. The difference in expression levels (abundance) of these two mRNAs is now translated to the difference in the intensities on the two images.

An example of a microarray data set is listed below. This data is collected from an HIV experiment for illustration. It has genetic expression of Peripheral Blood Mononuclear Cells (PBMC) from HIV positive humans dyed in one colour and genetic expressions of PBMC from HIV positive monkeys dyed in another colour. It contains two parts and we have access to the first part of the data, which has 9358 genes. Each of 9132 genes (DNA sequence) is attached on two spots, and each of the other 226 genes is attached on 4 or 6 spots. There are also some housekeeping genes on

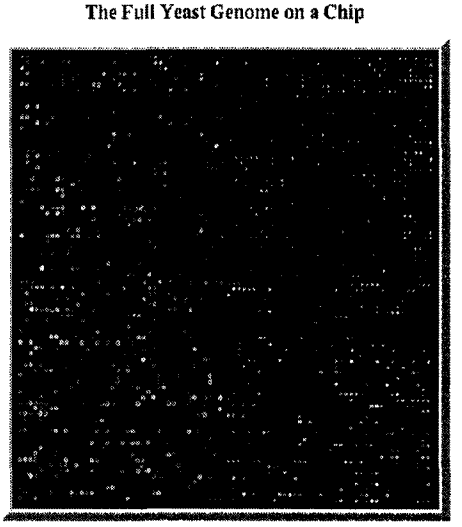

Figure 5: Microarray slide example. (From internet http://industry.ebi.ac.uk/~alan/MicroArray/IntroMicroArrayTalk by A.J. Robinson. Originally from DeRisi et.al. (Science 278: 680 686)

each slide and we will exclude them from our consideration. Three replicate slides are produced and the dye colours are flipped in one of the slides. The intensity of each gene on every spot for each color is measured. Part of the data is listed here for illustration (table 1).

2.3 Experimental Design

The statistical principles underlying experimental design are replication, randomization, and blocking. Replication makes it possible to estimate experimental error. The more replicates, the more accurate is the estimator. The probability that the hybridization of any single spot containing complementary DNA will not reflect the presence of the mRNA is about 5%, and the probability a spot can provide a substantial signal even if the corresponding mRNA is not present is about 10%([10], page 195). Therefore replication is very important and necessary in microarray experiments to provide a reliable estimator. However, more replication will cost more. Since microarray experiments deal with thousands of genes simultaneously, even adding one

gene	X Location1	Y Location1	Rdata1a	Rbdata1a	Gdata1a	Gbdata1a
R11726	1540	2890	559.810791	178.947372	1175.157104	376.236847
R11726	1730	2910	536.679016	165.10527	1264.101685	359.236847
R37412	1900	2910	772.121765	168.39473	1568.447144	225.868423
R37412	2060	2900	730.046753	144.236847	1824	342.868408
⋮	⋮	⋮	⋮	⋮	⋮	⋮
H68793	18340	38710	1471.555542	115.315788	2059.646484	144.578949
H68793	18500	38710	1345.82605	125.552635	1889.042725	165.947372
H72486	18680	38710	699.666687	145.815796	1011.455444	237.763153
H72486	18840	38710	785.470581	126.736839	908.836548	224.763153
gene	X Location2	Y Location2	Rdata2a	Rbdata2a	Gdata2a	Gbdata2a
R11726	1590	2990	609.073181	175.552628	880.876404	230.39473
R11726	1760	2990	845.844849	314.289459	1026.802368	337.394745
R37412	1940	3000	774.742432	233.447372	1309.761475	344.368408
R37412	2110	3000	761.050659	252.131577	1584.677734	400.605255
⋮	⋮	⋮	⋮	⋮	⋮	⋮
H68793	18510	38810	1125.413086	234.157898	1873.276733	187.052628
H68793	18690	38800	1008.359985	221.736847	791.784485	161.131577
H72486	18870	38810	846.486511	213.184204	792.814148	132.342102
H72486	19040	38800	463.038452	192.289474	341.940002	152.184204
gene	X Location3	Y Location3	Rdata3a	Rbdata3a	Gdata3a	Gbdata3a
R11726	3440	3620	613.302612	192.526321	1962.490601	760.447388
R11726	3610	3630	602.855103	176.289474	2171.142822	798.736816
R37412	3780	3640	654.435913	116.052635	3049.305664	960.394714
R37412	3920	3630	638.322021	69.789474	2642.386475	791.078918
⋮	⋮	⋮	⋮	⋮	⋮	⋮
H68793	20340	39270	3565.079346	89.421051	2284.054932	344.894745
H68793	20500	39270	3409.746094	134.921051	1894.378174	359.631592
H72486	20680	39250	2747.589844	115.631577	1401.654541	474.157898
H72486	20850	39280	3136.142822	116.368423	1097.333374	319.342102

Table 1: Illustration of microarray data from a HIV experiment. Three slides were performed. The human sample was dyed green and the monkey sample was dyed red in slide 1, and the colour is reversed in slides 2 and 3. The columns Rdata and Gdata are gene intensities and columns Rbdata and Gbdata are background measures.

more replicate for each gene, thousands more spots are actually added in the experiment. Thus, it is important to determine the suitable number of replicates for each experiment, which in turn is a sample size question. This thesis will address this question and provide a tool to calculate a reference sample size.

Randomization requires the factors that are not of interest but influence the outcome of the experiment to be randomly allocated in the experiment([10], page 196). One example is that if there are replicates within a slide, they should be printed at random locations throughout the slide, so that the gene effect will not be confounded with location effect, if there is any.

In a spotted microarray experiment, each spot is always a block. Usually each slide can also be treated as a big block and it is expected that the measurements of spots from a single slide will be more homogeneous than the measurements across slides. When genes of interest and replicates are across slides, block effects need to be considered, or careful normalization needs to be performed for data obtained from spots across slides.

When more than two conditions are compared in a microarray two-dye system, how to assign two dyes to those conditions so as to achieve more information on interesting comparisons is an issue for discussion. Typical experimental designs used are the reference design and the loop design (Figure 6).

A reference design uses a common reference sample for each slide, and the comparison between two treatments is through the differences of those treatments from the reference condition. Usually the reference condition is colored in one dye and all test conditions are colored with the other dye. Thus the dye effect is confounded with the condition effects. The advantage of the reference design is that it can be extended by adding additional slides: whenever extra experiments are performed for existing or new treatments, the data can be combined with the original data sets to obtain a bigger experiment size. The limitation of a reference design is that it always collects the most information on the reference condition, a case that is not of interest.

Kerr et al. [21] suggested using a loop design - the first and second condition are compared on the first slide, the second and third on the second, and so forth, and the same conditions are labeled different colors in consecutive slides. The loop design

does not need a reference group, thus double the amount of data on the interesting test conditions for the same number of slides. However a loop design might cost more since each sample must be labeled in both red and green florescent colors. When the number of conditions of interest is larger than three, the loop design will not provide direct comparisons between all pairs of conditions. This can be improved by adding direct comparisons between conditions that are not neighbors on the loop but of scientific interest. Some other designs like the 2*2 Latin square design are sometimes used. As Kerr et al. [21] have stated, microarray designs with the two-dye system with more than two conditions under study are always incomplete block designs and more research can be done to find efficient experimental plans addressing the questions of interest. They also suggested that each condition should be always labeled with both dyes to detect gene-specific dye effects if there are any.

Figure 6: (a) reference design, (b) loop design. V represent conditions of interest, R is a reference condition, and conditions connected by arrow are printed on the same slides. The condition in the arrow end is always dyed red and in the arrow head is dyed green.

2.4 Statistical Methods

2.4.1 Normalization

For almost all microarray data, before analyzing the data, the first step is to check the validity of the data and then to normalize them if necessary. Normalization does not refer here to any statistical process, but rather to a numeric calibration of the data so as to adjust the raw data for known systematic effects. These steps are analogous to adjusting the values from separate samples for potential deterministic differences

in the measurement tools. Normalization techniques rely heavily on the expectation that the vast majority of the genes on the microarray do not express differentially. Consequently, for the bulk of the data, there should not be any difference from slide to slide.

Currently there are no established standards for how the raw data should be normalized although some are emerging, eg. MIAME [4]. One can start by visually checking the colour map of the experiment (see Figure 5). A successful experiment is supposed to produce a slide with balanced colour and uniform hybridization in major spots, uniform spot distribution, and uniformly low background. Dust and scratches on the slide should be limited. Scatter plot, histograms, boxplots and spatial plots can be used to investigate if there are any specific experimental problems. For example, if the experiment is color-balanced, the scatter plot of log(Green) to log(Red) intensities should appear to be generally a straight line since only few genes are expected to be differentially expressed - most are expected to be equally expressed in both colours. On the other hand, if there is a strong dye effect, spots from even non-differential genes will nonetheless show high values in one dye and low values in the other dye and there will be a lot of points lying far from the straight line. In another process, the spots can be clustered as different groups according to their location on the slide and the boxplots of the log(green/red) per spot cluster will indicate skewed spot information if any exists. Such spatial effects can arise from the use of robots with a fixed low number of print tips to paint a large number of spots onto a slide.

Normalization should be done to correct for systematic differences between samples on the same slide, or between slides, which do not represent true biological variation between samples. For instance, dye biases are found to vary with overall spot intensity, location on the array, and scanning parameters. It is necessary to normalize the raw data to eliminate the bias. Genes to be used for normalization can be all genes on the array, housekeeping genes only, or control genes. A common method is to use all genes on the array. It is easy to perform but it assumes that only a relatively small proportion of the genes are expressed. Housekeeping genes are genes that have constant expression most of the time. The limitation of using housekeeping genes is that they may not be representative of genes of interest since they are always

highly expressed. Control genes are genes totally different from the genes of interest, perhaps genes from a completely different organism from the one under study. They are spotted on the array and also included in the two DNA samples at equal amounts for control purposes.

Normalization algorithms include array normalization (e.g, all values are divided by the mean of the array from which they were collected, or among only control spots/genes), background normalization (e.g., the intensity of the background is calculated from a local area around that spot or from all spots in a sub-grid area), and color normalization (a flip dye experiment can be conducted and then color distortion can be corrected through curve fitting on different color intensities for same genes).

Intensity data are frequently log_2-transformed as part of the normalization step. It is usually the case that a histogram of intensities for a single slide is heavily skewed, and that a *log* transformation produces a more symmetric distribution.

After the validation check and any normalization algorithms are executed, the microarray data is deemed ready for content analysis. This thesis will not pursue whether or not these normalization steps are valid or likely to be successful, but will accept them as being correct and having corrected for the systematic effects they have targeted. The main methods used to analyze microarray data content can be grouped into two categories: descriptive and inferential. A typical descriptive method is clustering.

2.4.2 Clustering Methods

A cluster method groups genes using their expression levels and defines clusters such that those genes within each cluster are more closely related to one another than genes from different clusters, according to some predefined distance measure (e.g. the correlation coefficients, the Euclidean distance, or Manhattan distance) between the data vectors associated to genes. Typical cluster methods are K-means clustering and hierarchical clustering. To perform K-means clustering, the number of clusters K is predefined. K points are arbitrarily chosen to represent the centres of clusters. Second, for each gene, the distances between it and the K cluster centres are calculated

and the gene is assigned to the closest cluster. In this way, genes are partitioned into K mutually exclusive and exhaustive groups. Third, the new centre of each cluster is calculated (e.g. the mean of the gene intensities for each cluster) and the second step is repeated so genes will possibly be reallocated to form a new set of clusters. These steps are repeated and genes are iteratively reallocated until some stopping criterion is met (e.g. the cluster centres have been found such that no genes move from one cluster to another in a subsequent step). The K-means method is easy to perform but it has several limitations. One is that the number of clusters K is subjectively defined. Another is if the initial centers are chosen differently, the resulting allocation of the genes might be different. Therefore, the cluster quality needs to be assessed before the result can be used. Different measurements, such as comparing the size of the clusters to the distance to the nearest cluster, can be used to assess the quality.

Hierarchical clustering, which has a tree structure, avoids these shortcomings. It includes bottom up or top down clustering. The bottom up method starts with leaves. Each gene is a cluster. The distance between each pair of clusters is calculated and the closest two clusters are merged together to form a new cluster. This step is repeated until only one cluster is left. The top down method is the opposite, which is similar to growing a tree from the root. It starts with one cluster that includes all genes, and it splits the cluster into two using some distance measure. The process is repeated until each cluster only contains one gene. The top down method is usually faster than the bottom up, but the clusters produced tend to reflect less accurately the structure present in the data. The hierarchical clustering provides inference about the relationship among genes in the same cluster and also between clusters through the tree structure([10], page 289).

Clustering always leads to readily interpretable figures and can be helpful for identify patterns. However, genes can always be clustered whether there exist any patterns or not, and the clustering results are always highly dependent on the distance measure used. More repeated experiments are necessary to reach a reliable conclusion from clustering. To accelerate calculation speed, usually the uninformative genes are filtered out before clustering.

2.4.3 Inferential Methods

Fold change

Early analyses of these types of data looked at the ratio (red/green) of the fluorescence intensity between dyes for each spot, which used to describe the relative abundance of each specific gene in these two mRNA samples. The ratio is compared to an arbitrary cut off point, like 2. If the ratio is greater than 2, then it is concluded that this gene is judged to be differentially expressed for these two mRNA samples. Since log transformed values are more easily interpretable and more meaningful from a biological point of view, log transformation of dye intensity was also used so that the differences between log intensities described the relative abundance. However, both fold change or log transformed fold change methods are simple but not precise, and they are sensitive to variance heterogeneity across genes.

Unusual Ratio

The fold change method was updated to the unusual ratio - a gene is considered differentially expressed if the ratio (red/green) is a certain distance from the mean ratio. The distance typically used is $\pm 2\sigma$, where σ is the standard deviation of the ratio distribution. The unusual ratio method adjusts for heterogeneity across genes since it uses thresholds related to the distribution of the ratios. However, there are always 5% of genes that are detected as differentially expressed no matter how many genes are really expressed.

Hypothesis Testing

The standard t-test was also applied separately by gene to the log ratios to test for differential expression. It was robust to variance heterogeneity across genes but had low power and unstable estimated gene specific variance due to the usually small number of replications of slides. To limit the effect of this shortcoming, one modification was to use the pooled variance across all genes [7], but then the test is again sensitive to variance heterogeneity. Another modification was to add a small constant to the denominator of the gene specific t-test to stabilize the scores [13].

Since huge numbers of comparisons (let m be the number of genes) are performed simultaneously, the probability of at least one Type I error increases dramatically. To control the overall probability of making a Type I error, some multiple test corrections need to be applied to the separate by-gene p-values. The traditional methods [18] are the Bonferroni correction ($\alpha_c = \alpha_F/m$) and Sidak correction ($\alpha_c = 1 - (1 - \sqrt[m]{\alpha_F})$). These methods are too conservative and the type I error for a single gene test α_c is too small for practical use no matter how big the type I error of the whole family α_F is, due to the large m. The Sidak single step adjustment method was less conservative. It adjusts the p-values to $\tilde{p}_i = 1-(1-p_i)^m$ and compare $\tilde{p}_i$ to α to make the decision. These methods all control the probability of making at least one Type I error - the familywise error rate (FWER). For very small p-values, there is apparently little difference between these methods [31].

Other alternative approaches for multiple testing, such as the False Discovery Rate (FDR) or the step-down method are frequently used. The FDR method was proposed by Benjamini and Hochberg [3], which tolerates some Type I errors but controls the expectation of false discovery rate and is less conservative. It makes decisions about the number of rejections based on comparing the ordered p-values with some cut-off points. The cut-off points are related to the order of the p-value. It estimates the expected value of (# of falsely rejected / # of hypotheses rejected in total). The Westfall-Young step down method adjusts p-values based on observed ordered p-values. Details will be described later in Chapter 5. Dabrowski [8] suggested a method of using the likelihood test on the ordered p-values to test that at least k genes are expressed. The details will be discussed further in the following section.

Global ANOVA Model

The usual ANOVA model can be applied to microarray data for a single gene when more than two conditions are to be compared. A global ANOVA model treats all the data from all the genes in a single large ANOVA structure. Kerr et al. (2001) described this direction in more detail and it will be summarized in a later section. The ANOVA model uses the original intensity instead of log ratios and depending on experimental interest, a fixed model or mixed effects model can be applied. A fixed

model is simple while a mixed model accounts for multiple sources of variation. Kerr et al. (2001) claimed that the ANOVA model accounts for each source of variance once they are included in the model so that normalization is not necessary any longer. However, the use of an ANOVA model requires a careful design to ensure a sufficient number of degrees of freedom to estimate the error, and makes rather strong joint normality assumptions over the entire array.

Bootstrap

For genetic data, frequently the replicate number is limited, and the accuracy of the estimate is questionable. Sometimes the distribution of the residuals is so heavily skewed, the ANOVA model cannot be used reliably to make statistical inferences about detecting differentially expressed genes. The bootstrap provides a solution to these problems. The bootstrap can be used to obtain confidence intervals for estimates of differential expression for individual genes. It can also be combined with clustering methods on gene data analysis and evaluate the reliability of clustering results.

The bootstrap provides a direct computational tool for assessing statistical accuracy by sampling from the original data. It is available no matter how mathematically complicated the estimate may be. According to the resampling method, the bootstrap can be divided into the nonparametric bootstrap and the parametric bootstrap. The nonparametric bootstrap method is used when no mathematical model is available. This approach is based on the empirical distribution of the original data by randomly sampling from the original data set with replacement in an iid manner, each sample the same size as the original set. This is done a large number of times, say n. Then an estimate is calculated for each of these n data sets to examine its behavior over these n replications. For example, if the accuracy of the standard deviation (SD) of a sample $(X_1, X_2, \cdots, X_{50})$ is the question of concern, then resample the data set with replacement 1000 times and calculate the standard deviation for each of them. These 1000 standard deviations form an estimated distribution for the SD of the original data set and the variance of the estimated distribution can represent the accuracy of the original SD. However, if we know that the original sample $(X_1, X_2, \cdots, X_{50})$ came from a normal distribution $N(0, \sigma^2)$, and σ^2 is unknown, we can resample 1000

size 50 data sets from $N(0, SD^2)$, and using the estimated distribution of SD from these 1000 samples to predict the accuracy of SD. This method depends on the specific parametric model and is called a parametric bootstrap. In general, a parametric bootstrap is available when the feature to be estimated is a component or a function of the parameter in a known family of distributions.

The bootstrap method is based on the assumption that for large n the empirical distribution converges to the original true distribution, so that the variability of the bootstrap estimate around the estimate from the original data set is expected to be similar to the variability of the estimate from the original data around the true parameter. Therefore, more attention should be paid when applying the bootstrap to a data set with outliers or a large number of missing values.

More details can be found in references: Brown & Botstein [5], Draghici [10], Hastie, Tibshirani, & Friedman [15], Kerr & Churchill [20], Dorman [32].

2.4.4 Key Papers

We summarize a few papers in this subsection to illustrate the popular methods of statistical analysis on microarray data.

Kerr, Martin, and Churchill [19]

"Analysis of variance for gene expression microarray data"

This paper applied the global ANOVA model (a global ANOVA model treats all the data from all the genes in a single large ANOVA structure) to microarray data to detect gene expression. It included several sources of variation in the model for adjustment: array effects(A), dye (D), variety (V), gene (G), interactions of array and gene (AG), and variety and gene (VG) effects. Kerr states that including A, D, and V effects "effectively normalize the data without the need to introduce preliminary data manipulation". The $(VG)_{kg}$ effect is the effect of interest and differential gene expression can be detected through non-zero differences of this effect across varieties for a given gene.

This paper considered the Latin square design and the more common reference

Table 2: Latin Square design and Reference design used in Kerr et al. (2001) paper.

	Latin Square Design		Reference Design	
dye	1	2	1	2
Red	liver	Muscle	Placenta	Placenta
Green	Muscle	liver	liver	Muscle

design (Table 2) on two arrays as an illustration. A mRNA sample from liver tissue was compared to a sample from muscle tissue. In the Latin square design, the liver sample was assigned to red and the muscle sample to green in one array, and reversed in color in the other array. DG and AVG effects were not considered since they were completely confounded with each other and were orthogonal to all other effects, which in consequence did not alter the estimates of other terms if left out of the model. A model with effects A, D, V, G, AG, and VG was applied to this data and gene effects G_g and interactions $(VG)_{kg}$ were estimated. The difference $(VG)_{1g} - (VG)_{2g}$ for each specific gene could be estimated by a simple formula.

In the reference design, an extra placenta sample was used with red dye in each array and only one of muscle and liver samples were compared to it on any single slide. Here the varieties and dyes effects are completely confounded so dye effects were not included in the model. Also AG effects were not included so some degrees of freedom were saved to estimate the error. However, since the AG effects are partially confounded with the VG effects in this design, excluding AG from the model leads to potentially biased estimates of VG effects. A model with effects A, V, G, VG was applied and the difference $(VG)_{1g} - (VG)_{2g}$ was estimated.

To determine if the differences $(VG)_{1g} - (VG)_{2g}$ are significantly different from zero and to avoid the normal distribution assumption, this paper used the parametric bootstrap method to estimate the confidence intervals of the difference: resample with replacement from the rescaled empirical distribution of the residuals from the original fit, add them to the fitted value from the original fit to generate a new set of observations. The model was fitted to 20,000 such bootstrap data and the 0.5th and 99.5th percentile of the 20,000 difference $(VG)_{1g} - (VG)_{2g}$ estimates were used

as the lower and upper bound for its 99% confidence interval.

The two designs were compared using two approaches: comparing only duplicates on each single slide or comparing common genes across all four arrays. Both results agree for most of the genes. In general, reference designs are readily extended but more and more data are collected on the uninteresting reference sample, i.e., the placenta alone. A Latin square design is better since it has narrower confidence intervals than reference designs. As Kerr & Churchill state, "In general, for a given number of arrays, designs that are balanced across the samples of interest will provide the greatest efficiency".

Lee and Whitemore (2002) [25]

"Power and sample size for DNA microarray studies"

As its name indicates, this paper focuses on power and sample size calculation for different experimental designs and aims to find designs with high power while controlling type I error. An ANOVA model with interaction terms is still applied and selected interaction parameters are used to measure differential expression of genes across experimental conditions.

This paper using a fixed ANOVA model to test the null hypothesis $H_0 : (VG)_g = 0$ (gene g is not differentially expressed) vs $H_a : (VG)_g = (VG)_d$ (gene g is differentially expressed with difference $(VG)_d$), under assumptions that $(VG)_g \sim N_d(0, \sigma^2)$ under H_0, and $(VG)_g \sim N_d((VG)_d, \sigma^2)$ under H_a. Here the $(VG)_g$ is the vector of interaction parameters for gene g. Instead of using each individual interaction effect under control or treatment group, the summary measures $S_g = h((VG)_g)$ are used for analysis, where $S_g \sim f_0(s)$ under H_0, and $S_g \sim f_1(s)$ under H_a are assumed. The function h can be any pre-defined function that captures the particular differential expression features. In this paper, a linear summary and a quadratic summary of differential expression are considered. The Sidak and Bonferroni multiple test corrections are used. The power calculation procedure was developed for each summary function for each gene, and for independent estimation error and dependent estimation error, respectively. Its application to matched treatment and control pairs designs, to complete randomized designs, and to isolated effect design (differential gene expression

Table 3: Multiple Testing Framework

test hypothesis	test declaration		total
	Not-Significant	Significant	
True	U	V	m_0
False	T	S	m_1
total	m-R	R	m

exist in one of C experiment conditions, but without knowledge of which condition it is) were illustrated. Also power assessment was performed on two pilot studies.

Benjamini, Y. and Hochberg, Y (1995) [3]

"Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing"

This paper suggested a new multiple test procedure - the False Discovery Rate (FDR). Instead of controlling the family-wise error rate as do the classical multiple test methods, this procedure controlled the expected proportion of errors among the rejected hypotheses, which was termed FDR. It was less conservative than the traditional multiple test methods and it tolerates some Type I errors, provided their number is small in comparison to the number of rejected hypotheses.

The FDR was defined as follows: consider the problem of simultaneously testing m null hypotheses H_1^0, H_2^0,$\cdots$, H_m^0, of which m_0 are true (see Table 3). Parameters m_0 and m_1 (the number of false null hypotheses) are usually unknown. Denote the number of hypotheses that have been rejected as R. R is random but observable.

Define the proportion of the rejected null hypotheses which are erroneously rejected as $Q = \frac{V}{R}$. Then the FDR is the expectation of Q (denoted as $E(Q)$).

The FDR is equivalent to the control rate of the classical multiple test - the Family-Wise Error Rate (FWER, which is equal to $P(V \geq 1)$) if all null hypotheses are true, and it is smaller than or equal to FWER when $m_0 < m$.

The paper defined the FDR multiple testing procedure as follows: Consider testing $H_1, H_2, \cdots, H_m$ based on the corresponding p-values $p_1, p_2, \cdots, p_m$. Denote the ordered observed p-values as $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(m)}$, and denote the null hypotheses corresponding to $p_{(i)}$ by $H_{(i)}$. The FDR procedure is: let k be the largest i for which

$$p_{(i)} \leq \frac{i}{m} q^*, \tag{2.4.1}$$

then reject each $H_{(i)}$, $i = 1, 2, \cdots, k$.

The paper proved that for independent test statistics, the FDR procedure controls the FDR at q^*. The authors indicated that this method has greater power than FWER controlling methods such as Holm's step down procedure.

This paper performed a simulation study to compare the performance (the proportion of the false hypotheses which are correctly rejected) for the FDR methods and two other FWER controlling methods: the Bonferroni method and Hochberg's method. Simulations show that the FDR performs substantially better than the other two methods, in terms of the proportion of the false null hypotheses which are correctly rejected.

Dabrowski (2006) [8]

"Statistics for Comparison of Two Independent cDNA Filter Microarrays"

This paper deals with the high multiplicity of tests generated by a microarray experiment. Independence across genes is assumed and a parametric approach is developed to detect whether or not the k largest observed gene expression differences are real differences. This paper considers experiments with two filter microarrays - one from a control and one from a treated preparation. A filter microarray is the type of microarray where only a single wash is applied to each slide, so that the original intensity is used. For this type of microarray, the difference from the previous papers discussed here is comparable to that between paired and unpaired experiments. It is assumed that there are 2 dots on each slide for each gene, and the data have been cleaned and normalized to remove slide surface effects, inter-slide scaling effects and other similar problems.

This paper focused on the distribution of p-values from each single gene test and developed a multiple comparison procedure for testing whether or not at least $k+1$ genes in the pool of genes exhibit differential expression or if a pre-determined group of genes show a cumulative significant effect even if individually no single gene test has a small p-value. It is assumed that the observations are statistically independent, the means and variances are fixed for a single gene on a single slide, but may vary between genes and between control and treatment slides. Continuous densities are assumed to exist.

For each single-gene comparison, the usual two-sample statistical tests, like the t-test for normally distributed genes, can be used to detect the differential expression. The limitation is that only 2 duplicates are available to estimate the mean value of genes expression under each condition.

When all genes in the experiment are considered, to test the hypothesis K_0: *exactly k genes are differentially expressed* vs K_a: *at least $k+1$ genes are differentially expressed*, this paper proposed to build a likelihood ratio test on the density of p-values. It is assumed that p_g for testing of each single gene $H_0 : \mu_{x_g} = \mu_{y_g}$ vs $H_a : \mu_{x_g} > \mu_{y_g}$, has continuous density f_0 under H_0 and f_1 under H_a. If a t-test is used for a single gene comparison, then f_0 is the uniform distribution on $[0,1]$ and f_1 is monotone decreasing on $[0,1]$. Using a likelihood ratio test, K_0 is rejected when the test statistic $D = \prod_{g \in \{f_1(p_{r_g})>1, r_g>k\}} f_1(p_{r_g})$ is big, where r_g is the rank of the p-value for gene g. Note that the test statistic D is a function of f_1. If we assume the normal distribution to have variance 1 for the single gene test statistic, the $f_1(p_{r_g})$ is sigmoid but can be approximated by a straight line. The linear decreasing function $f_1 = 2(1-x)$ is used as an example for illustration. The null distribution of its test statistic D has a simple expression and can be easily simulated so that the test can be conducted for collected data.

Any single gene test that generates a uniformly distributed p-value under H_0 can be applied to this approach. If the above assumptions are not biologically tenable, another appropriate single gene test can be used to conduct the multiple comparison approach. This approach can also be used to compare the effects of more than two treatments conditions.

Dudoit et al (2002) [11]

"Statistical Methods for Identifying Differentially Expressed Genes in Replication cDNA Microarray Experiments"

This paper used a multiple test procedure to study the differential expression of genes in a cDNA microarray experiment. It first computed a test statistic for each gene, and then used adjusted p-values of the test statistics to conduct a multiple test procedure. No specific parametric distribution was assumed for the test statistics. A permutation procedure was used to estimate the adjusted p-values. According to the author, this procedure "strongly controls the family-wise Type I error rate and takes into account the dependence structure between the gene expression levels".

This paper first plotted the intensity ratio $M = log_2 R/G$ vs. the mean log intensity $A = log_2\sqrt{RG}$ on data from a self-self microarray experiment, in which two identical mRNA samples were labeled with different dyes and hybridized to the same slide. It concluded that the MA plot revealed more than the corresponding *log*-intensity plots in terms of identifying spot artifacts, and was useful for normalization purposes. It proposed normalization methods based on robust local regression to deal with intensity dependent dye biases. It also argued that normalization was important and could not always be addressed by relying on unverified modeling assumptions.

To detect differentially expressed genes, this paper computed a two-sample Welch t-statistic for each gene. Then it drew a Q-Q plot with the ordered test statistics against the quantiles of a standard normal distribution, to visually identify genes with 'unusual' test statistics. It also drew plots of t-statistics, their numerators, and denominators, against absolute expression levels, to visually examine the features of genes with large absolute t-statistics.

To detect differentially expressed genes analytically, a multiple test procedure - the Westfall-Young *step-down minP adjusted p-values* method - is used. The authors claimed that this procedure took into account the dependence structure between the test statistics. In this procedure, the adjusted p-values are defined by

$$\tilde{p}_{r_j} = max_{k=1,\cdots,j}\{Pr(min_{l\in\{r_k,\cdots,r_m\}}P_l \leq p_{r_k}|H_o^C)\},$$

where $H_o^C = \cap_{j=1}^m H_0^j$ (all null hypotheses in the family are true), P_l denotes the

random variable for the unadjusted p-value of the lth hypothesis H_l, and p_{r_k} is the r_kth ordered unadjusted p-values of the t-statistics.

In this paper, a permutation algorithm was used to estimate adjusted p-values and to avoid parametric assumptions about the joint distribution of the test statistics. The details of the permutation method were described in the paper. Since the joint null distribution of $P_1, \cdots, P_m$ still needed to be estimated to calculate the adjusted p-values, the additional resampling could be computationally intensive. For ease of computation, this paper used a multiple test procedure based on the *step-down maxT adjusted p-values*, which were defined in terms of the test statistics T_j themselves as

$$\tilde{p}_{r_j} = max_{k=1,\cdots,j}\{pr(min_{l\in\{r_k,\cdots,r_m\}}|T_l| \geq |t_{r_k}| \ |H_o^C)\},$$

where $|t_{r_1}| \geq |t_{r_2}| \geq \cdots \geq |t_{r_m}|$ denote the ordered observed test statistics.

The maxT p-values were equal to the minP p-values when the test statistics T_j were identically distributed, and it would lead to unbalanced adjustments ([31], page 50) in the non-identically distributed case.

The methods were applied on two microarray experiments. Both experiments used a reference design, with 8 treatment mice and 8 control mice against a pooling cDNA from the 8 control mice, on 16 slides. The experiments differed in a way such that the mice carried different gene types in both treatment and control groups. The adjusted p-values were much larger than the corresponding unadjusted p-values. The result produced by the adjusted p-values reflected the patterns seen in the Q-Q plots. In addition, the genes identified had meaningful biological links to the mouse types.

Three single slide methods were also applied to individual slides. They could only identify part of the significant expressed genes detected by the adjusted p-value method, but with a large number of false positives.

2.4.5 Work To Be Done

We will start with single gene tests for different designs: Balanced Incomplete Block (BIB) design, reference design, and loop design. The distributions of p-values for each single gene test under both null and alternative hypotheses will be studied and the specific expression of the distributions will be calculated.

Here we assume that the data we used is already normalized and color balanced. That is, the data from different slides can be combined directly without any color adjustment. Slides are considered as blocks in each design.

Two multiple test procedures, the FDR method and the Westfall-Young resampling procedure, will be considered. For both methods, it will be useful to relate the sample size (number of slides) to the probability of rejecting a certain number of null hypotheses under different gene combinations; this will be useful in experimental design. We will study these relationships here. For the FDR method, notions of power will be developed, and lower bounds will be produced for the probability of rejecting a certain number of hypotheses and for the power of the FDR method. This could be applied to compare design efficiency.

Under the assumption of independence, the Westfall-Young resampling procedure is coincides with the step-down Sidak method, which is conservative. Also, it is time consuming to apply the step-down Sidak method in experimental design. We will propose a way to speed up the simulation for the first few adjusted p-values and will develop their asymptotic distribution under different gene combinations. This could be used as a guide for sample size selection. We will also discuss a modified adjustment suggested by an anonymous reader.

Chapter 3

Tests Applicable To a Single Gene

In this chapter we look carefully at the data for a single gene among the thousands of genes available in a microarray experiment. For an idealized setting we obtain expressions for the density of p-values under the null and alternative hypotheses. These expressions are too complex to employ directly in Chapter 4, so we propose simple approximate boundaries that will allow further theoretical development.

3.1 The Model

Each gene will appear on different spots on a single slide and on different slides. In full generality one would model all of those effects, but here we assume that the slides already have been adjusted, data has been background-corrected if required, and the intensities obtained from each slide are already normalized. Thus the data from all slides have had systematic effects removed and are ready to be combined and compared directly. In keeping with current practice in microarray data analysis we assume that the normalization procedures have provided data without systematic variation.

We assume that b slides will be used to hybridize genes, the same set of genes appear on every slide, and each gene appears once on each slide. For each single gene, we have t treatment conditions to be compared, and each treatment appears on at

least 2 slides. For each single gene, we will build standard models which include treatment effects and will evaluate the F-test statistics for differential expression between treatments. The distribution of p-values of the F-test under H_0: *not differentially expressed* vs H_a: *differentially expressed* will be studied. We will consider three experimental designs popular in current practice: Balanced Incomplete Block (BIB) design, reference design, and loop design. As discussed in Chapter 1, the reference design is a commonly used design in microarray experiments. It is easy to use, but it collects most of its information on the reference group. The loop design collects double the information on treatments groups using the same number of arrays as in reference designs. However, when the number of treatments to be compared is large, the loop becomes big and it is hard to obtain an accurate comparison between treatments which are located far apart in the loop. Theoretically, the BIB design will solve this problem, so we will consider it here also.

In the case that genes are spotted more than once on a single slide, the intensities collected could be treated as replicates and we will have more degrees of freedom to estimate the error terms and the estimators are more accurate. However, the relative efficiency of the designs is the same for any given number of replicates. Hence, we will assume no replication here for simplicity.

We will later develop a notion of power for False Discovery Method (FDR) and will use the distribution of independent p-values under alternative and null hypotheses for each gene to examine that notion, so the performance of the different designs can be compared.

3.1.1 Simple 2-way Model

We will employ the simple 2-way model

$$y_{ij} = \mu + B_i + T_j + e_{ij}, \tag{3.1.1}$$

where $i = 1, \cdots, b$, and $j = 1 \cdots t$. Here, y_{ij} denotes the observed intensity of the jth treatment on the ith slide. Each slide is treated as a block. B_i denotes the ith block effect, T_j is the jth treatment effect, and e_{ij} is the error term. Here we assume both effects B_i and T_j are fixed effects. The error terms are assumed to be independent,

and have a centered normal distribution with common variance $\sigma^2 > 0$. This variance may change from gene to gene, but is assumed constant across (normalized) slides for a single gene. In this model, recall that we assume that a careful normalization has already been performed on the intensity measures y_{ij} so that the intensities from different dyes can be compared on an equal basis.

The contrasts between treatment conditions $T_{j_1} - T_{j_2}$ are of interest here and will be used to evaluate whether a gene is differentially expressed under conditions j_1 and j_2.

Assuming n_{ij} is the number of observations for the ith block and jth treatment, denote $n_{i\cdot} = \sum_{j=1}^{t} n_{ij}$, $n_{\cdot j} = \sum_{i=1}^{b} n_{ij}$, and $n_{\cdot\cdot} = \sum_i \sum_j n_{ij}$. Then $n_{i\cdot} = 2$ for all $i = 1, \cdots b$, since each slide only contain 2 treatments.

The model can be written as $Y = X\beta$, where β is the parameter vector $(\mu, B_1, \cdots, B_b, T_1, \cdots, T_t)'$, Y is the $tb \times 1$ observation vector and X is the $tb \times (1 + b + t)$ design matrix. For notational simplicity, we will add rows with all 0 elements to X and Y to represent those pseudo observation for $n_{ij} = 0$. Ordering y_{ij} by slide and then by treatment, we will have

$$Y = \begin{Bmatrix} y_{11} \\ \cdots \\ y_{1t} \\ y_{21} \\ \cdots \\ y_{2t} \\ \cdots \\ y_{b1} \\ \cdots \\ y_{bt} \end{Bmatrix}_{bt\times 1}, \quad X = \begin{Bmatrix} & n_{11} & 0 & \cdots & 0 & n_{11} & 0 & \cdots & 0 \\ & n_{12} & 0 & \cdots & 0 & 0 & n_{12} & \cdots & 0 \\ & \cdots & & \cdots & & & & \cdots & \\ & n_{1t} & 0 & \cdots & 0 & 0 & 0 & \cdots & n_{1t} \\ & 0 & n_{21} & \cdots & 0 & n_{21} & 0 & \cdots & 0 \\ & 0 & n_{22} & \cdots & 0 & 0 & n_{22} & \cdots & 0 \\ U_{bt\times 1} & \cdots & & \cdots & & & & \cdots & \\ & 0 & n_{2t} & \cdots & 0 & 0 & 0 & \cdots & n_{2t} \\ & \cdots & & \cdots & & & & \cdots & \\ & 0 & 0 & \cdots & n_{b1} & n_{b1} & 0 & \cdots & 0 \\ & 0 & 0 & \cdots & n_{b2} & 0 & n_{b2} & \cdots & 0 \\ & \cdots & & \cdots & & & & \cdots & \\ & 0 & 0 & \cdots & n_{bt} & 0 & 0 & \cdots & n_{bt} \end{Bmatrix}$$

$U_{bt\times 1}$ is a vector with element 0 for $n_{ij} = 0$ and element 1 elsewhere. For $n_{ij} = 0$, the corresponding $y_{ij} = 0$.

Therefore, the matrix $X'X$ is as follows:

$$X'X = \begin{Bmatrix} n_{..} & n_{1\cdot} & n_{2\cdot} & \cdots & n_{b\cdot} & n_{\cdot 1} & n_{\cdot 2} & \cdots & n_{\cdot t} \\ n_{1\cdot} & n_{1\cdot} & 0 & \cdots & 0 & n_{11} & n_{12} & \cdots & n_{1t} \\ n_{2\cdot} & 0 & n_{2\cdot} & \cdots & 0 & n_{21} & n_{22} & \cdots & n_{2t} \\ \cdots & & & \cdots & & & & \cdots & \\ n_{b\cdot} & 0 & 0 & \cdots & n_{b\cdot} & n_{b1} & n_{b2} & \cdots & n_{bt} \\ n_{\cdot 1} & n_{11} & n_{21} & \cdots & n_{b1} & n_{\cdot 1} & 0 & \cdots & 0 \\ n_{\cdot 2} & n_{12} & n_{22} & \cdots & n_{b2} & 0 & n_{\cdot 2} & \cdots & 0 \\ \cdots & & & \cdots & & & & \cdots & \\ n_{\cdot t} & n_{1t} & n_{2t} & \cdots & n_{bt} & 0 & 0 & \cdots & n_{\cdot t} \end{Bmatrix}$$

The matrix $X'X$ has order $1 + b + t$. The sum of the b columns after the first equals the first column, and the sum of the last t columns equals the first column also. Thus, the rank of $X'X$ is $b + t - 1$. Therefore, the normal equations $X'X\beta = X'Y$ has no unique solution. The general form of the solutions could be described through the generalized inverse of $X'X$. If β^0 is a solution of the equations, then β^0 has the representation $\beta^0 = GX'Y$, where G is a generalized inverse of $X'X$.

Searle [29, page 215] described a procedure for deriving β^0 and G. We will follow this procedure to obtain a solution β^0 and a generalized inverse G for our model: Since $X'X$ has $1 + b + t$ rows and its rank is $b + t - 1$, we can delete 2 rows and the corresponding columns from $X'X$, to leave a symmetric sub-matrix (denoted $(X'X)_l$) of full rank. Corresponding to the rows deleted from $X'X$, we delete elements from $X'Y$, and call the modified vector $(X'Y)_l$. Then solve $\beta_l^0 = [(X'X)_l]^{-1}(X'Y)_l$. In β^0, all elements corresponding to rows deleted from $X'X$ are zero; other elements are those of β_l^0, in sequence. The generalized inverse G corresponding to the solution β^0 can be obtained in the following way: in $X'X$, we replace all elements of $(X'X)_l$ by those of its inverse, and put all other elements zero. The resulting matrix is G.

One of the easiest ways to do this for our model is to put $\mu^0 = 0$ and $T_t^0 = 0$. The

equations left are

$$\left\{\begin{array}{cccccccc} n_{1\cdot} & 0 & \cdots & 0 & n_{11} & n_{12} & \cdots & n_{1(t-1)} \\ 0 & n_{2\cdot} & \cdots & 0 & n_{21} & n_{22} & \cdots & n_{2(t-1)} \\ \cdots & & \cdots & & & & \cdots & \\ 0 & 0 & \cdots & n_{b\cdot} & n_{b1} & n_{b2} & \cdots & n_{b(t-1)} \\ n_{11} & n_{21} & \cdots & n_{b1} & n_{\cdot 1} & 0 & \cdots & 0 \\ n_{12} & n_{22} & \cdots & n_{b2} & 0 & n_{\cdot 2} & \cdots & 0 \\ \cdots & & \cdots & & & & \cdots & \\ n_{1(t-1)} & n_{2(t-1)} & \cdots & n_{b(t-1)} & 0 & 0 & \cdots & n_{\cdot(t-1)} \end{array}\right\} \left\{\begin{array}{c} B_1^0 \\ B_2^0 \\ \vdots \\ B_b^0 \\ T_1^0 \\ T_2^0 \\ \vdots \\ T_{t-1}^0 \end{array}\right\} = \left\{\begin{array}{c} y_{1\cdot} \\ y_{2\cdot} \\ \vdots \\ y_{b\cdot} \\ y_{\cdot 1} \\ y_{\cdot 2} \\ \vdots \\ y_{\cdot t-1} \end{array}\right\}$$

The left-most matrix will be denoted as $(X'X)_l$. We split it into submatrices as follows:

$$(X'X)_l = \left\{\begin{array}{cc} D_b & N \\ N' & D_{t-1} \end{array}\right\}$$

where

$$D_b = \left\{\begin{array}{cccc} n_{1\cdot} & 0 & \cdots & 0 \\ 0 & n_{2\cdot} & \cdots & 0 \\ 0 & 0 & \cdots & n_{b\cdot} \end{array}\right\}, \quad D_{t-1} = \left\{\begin{array}{cccc} n_{\cdot 1} & 0 & \cdots & 0 \\ 0 & n_{\cdot 2} & \cdots & 0 \\ 0 & 0 & \cdots & n_{\cdot t-1} \end{array}\right\}$$

and

$$N = \left\{\begin{array}{ccc} n_{11} & n_{12} & \cdots n_{1(t-1)} \\ n_{21} & n_{22} & \cdots n_{2(t-1)} \\ \cdots & & \cdots \\ n_{b1} & n_{b2} & \cdots n_{b(t-1)} \end{array}\right\}.$$

Using the statement that

$$\left\{\begin{array}{cc} A & B \\ B' & D \end{array}\right\}^{-1} = \left\{\begin{array}{cc} A^{-1} + A^{-1}BWB'A^{-1} & -A^{-1}BW \\ -WB'A^{-1} & W \end{array}\right\},$$

where $W = (D - B'A^{-1}B)^{-1}$, we can obtain the inverse of $(X'X)_l$:

$$\begin{aligned} W &= (D_{t-1} - N'D_b^{-1}N)^{-1} = C^{-1}, \\ (X'X)_l^{-1} &= \left\{\begin{array}{cc} D_b^{-1} + D_b^{-1}NC^{-1}N'D_b^{-1} & -D_b^{-1}NC^{-1} \\ -C^{-1}N'D_b^{-1} & C^{-1} \end{array}\right\} \end{aligned}$$

To solve $(X'X)_l\beta_l^0 = X_l'Y_l$, we split $X_l'Y_l$ into

$$X_l'Y = \left\{ \begin{array}{c} y1 \\ y2 \end{array} \right\},$$

where

$$y1 = \left\{ \begin{array}{c} y_{1\cdot} \\ \vdots \\ y_{b\cdot} \end{array} \right\}, \quad y2 = \left\{ \begin{array}{c} y_{\cdot 1} \\ \vdots \\ y_{\cdot t-1} \end{array} \right\}.$$

Then a solution to the normal equations is

$$\beta^0 = \left\{ \begin{array}{c} 0 \\ B^0 \\ T^0 \\ 0 \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ D_b^{-1}y1 - D_b^{-1}NT^0 \\ C^{-1}(y2 - N'D_b^{-1}y1) \\ 0 \end{array} \right\}.$$

Denote $V = y2 - N'D_b^{-1}y1 = \{v_j\}$. Then we have $v_j = y_{\cdot j} - \sum_{i=1}^{b} \frac{n_{ij}}{n_{i\cdot}} y_{i\cdot}$, for $j = 1, \cdots, t-1$. For $C = \{c_{jj'}\} = D_{t-1} - N'D_b^{-1}N$, we have $c_{jj} = n_{\cdot j} - \sum_{i=1}^{b} \frac{n_{ij}^2}{n_{i\cdot}}$ for $j = 1, \cdots, t-1$, and $c_{jj'} = -\sum_{i=1}^{b} \frac{n_{ij}n_{ij'}}{n_{i\cdot}}$, for $j' \neq j$.

Denote by $\bar{y}_b$ the vector

$$\bar{y}_b = \left\{ \begin{array}{c} \frac{y_{1\cdot}}{n_{1\cdot}} \\ \frac{y_{2\cdot}}{n_{2\cdot}} \\ \vdots \\ \frac{y_{b\cdot}}{n_{b\cdot}} \end{array} \right\},$$

and so we have $\bar{y}_b = D_b^{-1}y1$. Then the solution β^0 can be written as

$$\beta^0 = \left\{ \begin{array}{c} 0 \\ B^0 \\ T^0 \\ 0 \end{array} \right\} = \left\{ \begin{array}{c} 0 \\ \bar{y} - D_b^{-1}NC^{-1}V \\ C^{-1}V \\ 0 \end{array} \right\}.$$

The generalized inverse of $X'X$ corresponding to this solution is

$$G = \left\{ \begin{array}{cccc} 0 & 0 & 0 & 0 \\ 0 & D_b^{-1} + D_b^{-1}NC^{-1}N'D_b^{-1} & -D_b^{-1}NC^{-1} & 0 \\ 0 & -C^{-1}ND_b^{-1} & C^{-1} & 0 \\ 0 & 0 & 0 & 0 \end{array} \right\}.$$

Our goal of using this model is not to estimate a solution only, but to compare treatment effects through hypothesis testing. To test a testable linear combination of parameters $H : S'\beta = s$, denote $Q = (S'\beta^0)'(S'GS)^{-1}(S'\beta^0)$, then since

$$Y \sim N(X\beta, \sigma^2 I), \quad \beta^0 = GX'Y \sim N(GX'X\beta, \sigma^2 GX'XG'),$$

and by the fact that $H : S'\beta = s$ is testable, we have

$$S'\beta^0 \sim N(S'\beta, S'GS\sigma^2), \quad Q/\sigma^2 \sim \chi^{2'}[\nu_1, \frac{1}{2\sigma^2}(S'\beta)'(S'GS)^{-1}(S'\beta)],$$

with $\nu_1 = rank(S)$, and where Q and SSE are distributed independently. Therefore,

$$\begin{aligned} F(H) &= \frac{Q/\nu_1}{SSE/(N-rx)} \\ &= \frac{(S'\beta^0)'(S'GS)^{-1}(S'\beta^0)/\nu_1}{SSE/(N-rx)} \\ &\sim F'(\nu_1, N-rx, (S'\beta)'(S'GS)^{-1}(S'\beta)/2\sigma^2) \\ &= F'(\nu_1, N-rx, s'(S'GS)^{-1}s/2\sigma^2) \ \textit{under} \ H. \end{aligned}$$

Here $rx = rank(X)$, N is the number of observations, $SSE/\sigma^2 = (Y'(I-XGX')Y)/\sigma^2 \sim \chi^2_{N-rx}$, and F' is a non-central F distribution.

In microarray experiments, slide effects are not of interest and contrasts between treatments effects are usually the focus of the experiments. Therefore, hypothesis tests are only related to T^0 when applying this model. The matrix S' always has the form $(0, 0, K)$. The first 0 is a vector zero, it contains the coefficient for μ; the second 0 is a matrix zero, it contains the coefficients for $B^0_1, \cdots, B^0_b$. Thus, $S'\beta = K'T$, and $S'GS = K' \begin{pmatrix} C^{-1} & 0 \\ 0 & 0 \end{pmatrix} K$.

For example, to test the hypothesis that all treatments are same vs all treatments are different from the treatment t (e.g, only one treatment t actually has effect on a disease) by a constant, the hypothesis is: $H_0 : T_j - T_t = 0$ vs $H_a : T_j - T_t = a$ for $j = 1, \cdots, t-1$ with a constant a. If the matrix notation is used, the hypothesis is

$H_0 : K'T = 0$ vs $H_a : K'T = a1_{t-1}$, where

$$K' = \begin{Bmatrix} 1 & 0 & \cdots & 0 & -1 \\ 0 & 1 & \cdots & 0 & -1 \\ & & \cdots & & \\ 0 & 0 & \cdots & 1 & -1 \end{Bmatrix}_{(t-1)\times t} \qquad 1_{t-1} = \begin{Bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{Bmatrix}_{(t-1)\times 1}.$$

Substituting these values into the test statistic expression gives

$$S' = (0\ 0\ K') = (0\ 0\ I_{t-1}\ -1_{t-1}), \quad S'GS = I_{t-1}C^{-1}I_{t-1} = C^{-1},$$

$$S'\beta = K'T, \quad S'\beta^0 = C^{-1}V,$$

$$Q = (S'\beta^0)'(S'GS)^{-1}S'\beta^0 = VC^{-1}V.$$

Thus, the test statistics for this hypothesis is

$$F(H) = \frac{Q/\nu_1}{SSE/(N-rx)} \sim F'(\nu_1, N-rx, (K'T)'C(K'T)/2\sigma^2).$$

It has a central F distribution under H_0, and a non-central F' distribution with non-centrality parameter $\delta = a^2 1'_{t-1}C1_{t-1}/2\sigma^2$ under H_a. Therefore, for large $F(H)$ (i.e. for small p-values), the null hypothesis will be rejected and the gene will be concluded to be differentially expressed.

Another commonly used hypothesis is $H_0 : K'T = 0$ vs $H_a : K'T = \{a_j\}$ - under the alternative, the differences between treatments differ. The test statistic for this hypothesis is same as the one above, but it has an F' distribution with non-centrality parameter $\delta = \{a_j\}'C\{a_j\}/2\sigma^2$ under the alternative.

Other comparisons between treatments $T_j - T_{j'}$ can be estimated by the difference of the estimated values of $T_j - T_t$ and $T_{j'} - T_t$, and their F-statistic can be calculated using the corresponding matrix K.

3.1.2 Other Models

There are other models which can be applied in microarray experiments. For example, if multiple spots are used for one gene, and if we also want to adjust for spot

differences, then the model $Y_{ijk} = \mu + B_i + T_j + S_{(i)k} + e_{ijk}$ can be used. Here, $S_{(i)k}$ is the kth spot effect nested in the ith slide. This model becomes more practical when the spots of replicates are located far from each other on the slides and/or have different locations on different slides. The analysis will be similar to that described in the preceding section. We will rename the BIB design the "modified BIB design" in this case, since now each block (slide) contains the same treatment more than once, while the regular BIB design requires that no treatment appears more than once in any block. However, the design is still balanced across slides and we cannot allocate more than two treatments on the same slide no matter how many spots are available due to the two dye system limitation in microarray experiments. This modification will be necessary for the definition of the BIB design when genes are multiply spotted on the same slide even without considering the spot effects in the model. However, this modification will not influence the method of analysis.

A dye effect can also be added to the model, or we can use a mixed model - i.e., treat some effects such as the slide effects as random. The procedure to obtain the estimators and the test statistic is similar to the preceding but more complicated. However, once the test statistic for treatment differences is determined and its p-values is calculated, the multiple testing procedures can be applied to detect the differentially expressed genes. We will discuss different multiple testing procedures and estimate the asymptotic distributions of the adjusted p-values in next few chapters. Model (3.1.1) will be used in this thesis since we will focus on experimental design and this model is adequate for this purpose. Other more complicated models could be applied for data analysis.

3.2 Non-centrality Parameters

In this section we will consider three different designs - the BIB design, the reference design, and the Loop design. We will use model (3.1.1) and compare the non-centrality parameters of test statistics for different designs under the alternative hypothesis $H_a : T_j - T_l = a_j$ for $j = 1, \cdots, t-1$. In general, if the non-centrality

parameter is larger, it will be easier to reject the null hypothesis and detect the differentially expressed genes, and the corresponding design is better. We assume b slides will be used to compare t treatments, and each treatment will appear at most once on each slide for each gene.

3.2.1 BIB Design

For each single gene, a balanced incomplete block (BIB) design is used to compare t treatments. Each microarray slide is considered as a block, and each block contains 2 treatment conditions. Assume we have b blocks, t treatments, each treatment appears r times, and each pair of treatments appears together λ times. We know for a BIB design, those parameters are not independent and they must satisfy the following restrictions:

$$N = t \times r = b \times 2, \qquad \lambda \times (t-1) = r, \qquad b \geq t.$$

To establish the second restriction, we consider any treatment T. It appears in r blocks, in each of which is also another treatment. The r blocks have to contain the remaining $t-1$ treatments exactly λ times each. Hence, $r = \lambda \times (t-1)$. For r and λ to be integers, it is equivalent to requiring that $2b/t$ and $2b/(t \cdot (t-1))$ be integers. Under these restrictions, the number of observations of treatment j in each block i satisfies the following equations:

$$n_{ij} = 0 \text{ or } 1, \quad n = n_{..} = \sum_i \sum_j n_{ij}, \quad n_{ij}^2 = n_{ij}$$

$$n_{i.} = \sum_j n_{ij} = 2, \qquad n_{.j} = \sum_i n_{ij} = r, \qquad \sum_i n_{ij} n_{ij'} = \lambda,$$

and $rank(X) = b + t - 1$. To find the distribution of the test statistics, we need to first calculate the matrix C. Plugging the restrictions of n_{ij} into the definition of the matrix C, we have

$$c_{jj} = n_{.j} - \sum_{i=1}^{b} \frac{n_{ij}^2}{n_{i.}} = r - \sum_{i=1}^{b} \frac{n_{ij}}{2} = \frac{r}{2}, \quad for \ j = 1, \cdots, t-1,$$

$$c_{jj'} = -\sum_{i=1}^{b} \frac{n_{ij}n_{ij'}}{n_{i\cdot}} = -\frac{1}{2}\lambda, \quad for \ j \neq j', \ j, j' = 1, \cdots, t-1.$$

Therefore,

$$C_{BIB} = \frac{1}{2} \begin{pmatrix} r & -\lambda & \cdots & -\lambda \\ -\lambda & r & \cdots & -\lambda \\ \cdots & \cdots & \cdots & \cdots \\ -\lambda & -\lambda & \cdots & r \end{pmatrix}_{t-1},$$

here, $r = 2b/t$ and $\lambda = 2b/(t \cdot (t-1))$.

To test $H_0 : K'T = 0$ vs $H_a : K'T = \{a_j\}$, where $K' = (I_{t-1} \ -1_{t-1})$, the test statistic $F(H) = \frac{Q/\nu_1}{SSE/(N-rx)} \sim F'(t-1, b-t+1, \delta)$ has a central F distribution under H_0 and a non-central F' distribution with non-centrality parameter

$$\begin{aligned} \delta &= (K'T)'C_{BIB}(K'T)/2\sigma^2 \\ &= \{a_j\}'C_{BIB}\{a_j\}/2\sigma^2 \\ &= \frac{1}{4\sigma^2} \cdot [(r+\lambda)\sum_{j=1}^{t-1} a_j^2 - \lambda(\sum_{j=1}^{t-1} a_j)^2] \\ &= \frac{1}{2\sigma^2} \cdot \frac{b}{t(t-1)}[t\sum_{j=1}^{t-1} a_j^2 - (\sum_{j=1}^{t-1} a_j)^2]. \end{aligned}$$

under H_a. For the special case $\{a_j\} = a1_{t-1}$, we have under H_a,

$$\begin{aligned} \delta &= \frac{1}{2\sigma^2} \cdot \frac{b}{t(t-1)}[t\sum_{j=1}^{t-1} a_j^2 - (\sum_{j=1}^{t-1} a_j)^2] \\ &= \frac{1}{2\sigma^2} \cdot \frac{a^2 b}{t}. \end{aligned}$$

3.2.2 Reference Design

For each single gene, a reference design is frequently used in practice - an extra treatment condition besides the t treatments is required. This condition is named the reference condition and is labeled with one color (e.g. red). Each of the t treatments is labeled in another color (e.g. green), and is hybridized to the slide together with the

reference condition, see Figure 6a (page 15). Thus, the reference condition appears in each slide and so it is repeated b times. For the other t treatment conditions, each slide only contains one of them. Assume each of the t treatments is repeated r times, then $r = b/t$, given that b/t is an integer.

The number of observations n_{ij} in this design has the following properties:

$$\begin{aligned} n_{ij} &= 0 \text{ or } 1, \quad n_{ij}^2 = n_{ij}, \quad n_{i\cdot} = 2, \\ n_{\cdot j} &= \begin{cases} r, & for\ j = 1 \cdots t, \\ b, & for\ j = t+1. \end{cases} \\ and\ \sum_i n_{ij} n_{ij'} &= \begin{cases} 0 & for\ j = 1, \cdots, t,\ and\ j' \neq t+1, \\ r & for\ j = 1, \cdots, t,\ and\ j' = t+1. \end{cases} \end{aligned}$$

Now $rank(X) = b + t$. In this design, all treatments of interest are compared to the reference treatment only, and there are $(t+1)$ conditions in total. Change t to $(t+1)$ in the model, and let $\mu = 0$ and $T_{t+1} = 0$ to solve the normal equations. The same expressions can be used to estimate the parameters, with the size of matrices (or vectors) N, C, and V be modified respectively - N has size $b \times t$, C has size $t \times t$, and V has size $t \times 1$. We will rename C as C_t to emphasize its size. The elements of C_t are:

$$c_{jj} = n_{\cdot j} - \sum_{i=1}^{b} \frac{n_{ij}^2}{n_{i\cdot}} = r - \sum_{i=1}^{b} \frac{n_{ij}}{2} = \frac{r}{2}, \quad for\ j = 1, \cdots, t,$$

$$c_{jj'} = -\sum_{i=1}^{b} \frac{n_{ij} n_{ij'}}{n_{i\cdot}} = 0, \quad for\ j \neq j',\ j, j' = 1, \cdots, t.$$

That is,

$$C_t = \begin{pmatrix} \frac{r}{2} & 0 & \cdots & 0 \\ 0 & \frac{r}{2} & \cdots & 0 \\ & \cdots & \cdots & \\ 0 & 0 & \cdots & \frac{r}{2} \end{pmatrix}_t .$$

To test $H_0 : K'T = 0$ vs $H_a : K'T = \{a_j\}$, where $K' = (I_{t-1} \ -1_{t-1} \ 0)$, (the 0 is for the reference condition,) we have

$$S' = (0 \ 0 \ K') = (0 \ 0 \ I_{t-1} \ -1_{t-1}, \ 0),$$

$$\begin{aligned} S'GS &= (I_{t-1} \ -1_{t-1}) \cdot C_t^{-1} \cdot \begin{pmatrix} I_{t-1} \\ -1'_{t-1} \end{pmatrix} \\ &= \frac{2}{r} \cdot (I_{t-1} + 1_{t-1}1'_{t-1}) \\ &= \frac{2}{r} \begin{pmatrix} 2 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & 1 \\ & \cdots & \cdots & \\ 1 & 1 & \cdots & 2 \end{pmatrix}_{t-1}, \end{aligned}$$

It is easy to calculate that

$$U = (S'GS)^{-1} = \frac{r}{2t} \begin{pmatrix} t-1 & -1 & \cdots & -1 \\ -1 & t-1 & \cdots & -1 \\ & & \cdots & \cdots \\ -1 & -1 & \cdots & t-1 \end{pmatrix}_{t-1}.$$

We have

$$\begin{aligned} S'\beta &= K'T, \quad S'\beta^0 = (I_{t-1} \ -1_{t-1})C_t^{-1}V, \\ Q &= (S'\beta^0)'(S'GS)^{-1}S'\beta^0 = V'C_t^{-1} \begin{pmatrix} I_{t-1} \\ -1'_{t-1} \end{pmatrix} U(I_{t-1} \ -1_{t-1})C_t^{-1}V \\ &= \frac{2}{tr}V' \begin{pmatrix} \frac{2t}{r}U & -1_{t-1} \\ -1'_{t-1} & t-1 \end{pmatrix} V \end{aligned}$$

Thus, the test statistic

$$F(H) = \frac{Q/\nu_1}{SSE/(N-rx)} \sim F'(t-1, b-t, \delta)$$

has a central F distribution under H_0 and a non-central F' distribution with non-centrality parameter

$$\begin{aligned}\delta &= (K'T)'U(K'T)/2\sigma^2 \\ &= \{a_j\}'U\{a_j\}/2\sigma^2 \\ &= \frac{1}{2\sigma^2}\cdot\frac{r}{2t}[t\sum_{j=1}^{t-1}a_j^2-(\sum_{j=1}^{t-1}a_j)^2] \\ &= \frac{1}{2\sigma^2}\cdot\frac{b}{2t^2}[t\sum_{j=1}^{t-1}a_j^2-(\sum_{j=1}^{t-1}a_j)^2]\end{aligned}$$

under H_a. Here, $\nu_1 = rank(K') = t-1$, and $N - rx = 2b - (b+t) = b - t$. For the special case $\{a_j\} = a1_{t-1}$, we have under H_a,

$$\begin{aligned}\delta &= \frac{1}{2\sigma^2}\cdot\frac{b}{2t^2}[t\sum_{j=1}^{t-1}a_j^2-(\sum_{j=1}^{t-1}a_j)^2] \\ &= \frac{1}{2\sigma^2}\cdot\frac{a^2b(t-1)}{2t^2}.\end{aligned}$$

3.2.3 Loop Design

A loop design (see Figure 6b, page 15) can be used for each single gene between treatments. Ordering the t treatments in a list $\{1, 2, \cdots, t\}$, the loop design requires that each treatment j is only hybridized with the preceding treatment $(j-1)$ and with the following treatment $(j+1)$. The 1^{st} treatment is considered as the treatment following the tth treatment (the last treatment), so that a loop is formed. Treatments that are not next to each other will not be hybridized in the same slides. For a loop design with b slides and t treatments, if each treatment appear r times, then r is even. Each pair of consecutive treatments will appear together $\frac{r}{2}$ times.

Using the n_{ij} notation, for $t > 2$, we have

$$n_{ij} = 0 \text{ or } 1, \quad n_{ij}^2 = n_{ij}, \quad n_{i\cdot} = 2, \quad n_{\cdot j} = r,$$

$$\text{and } \sum_i n_{ij}n_{ij'} = \begin{cases} \frac{r}{2} & if j' = j+1 \text{ or } j-1, \\ 0 & otherwise. \end{cases}$$

and $rank(X) = b+t-1$. Here $r = 2b/t$ is an integer, $j-1 = t$ if $j = 1$, and $j+1 = 1$ if $j = t$.

To calculate matrix C, plug these conditions of n_{ij} into the definition of C:

$$c_{jj} = n_{\cdot j} - \sum_{i=1}^{b} \frac{n_{ij}^2}{n_{i\cdot}} = \frac{r}{2}, \quad for \ \ j = 1, \cdots, t-1,$$

$$c_{jj'} = -\sum_{i=1}^{b} \frac{n_{ij}n_{ij'}}{n_{i\cdot}} = \begin{cases} -\frac{r}{4} & if \ \ (j > 1, j' = j+1 \ \ or \ \ j-1) \\ & \quad or \ \ (j = 1, j' = 2), \\ 0 & otherwise. \end{cases}$$

That is,

$$C_{loop} = \begin{pmatrix} \frac{r}{2} & -\frac{r}{4} & & & \\ -\frac{r}{4} & \frac{r}{2} & -\frac{r}{4} & & \\ & -\frac{r}{4} & \frac{r}{2} & -\frac{r}{4} & \\ & \cdots & \cdots & \cdots & \\ & & & -\frac{r}{4} & \frac{r}{2} \end{pmatrix}_{t-1},$$

with $r = 2b/t$.

To test $H_0 : K'T = 0$ vs $H_a : K'T = \{a_j\}$, where $K' = (I_{t-1} \ \ -1_{t-1})$, the test statistic

$$F(H) = \frac{Q/\nu_1}{SSE/(N-rx)} \sim F'(t-1, b-t+1, \delta)$$

has a central F distribution under H_0 and a non-central F' distribution with non-centrality parameter

$$\begin{aligned} \delta &= (K'T)'C_{loop}(K'T)/2\sigma^2 \\ &= \{a_j\}'C_{loop}\{a_j\}/2\sigma^2 \\ &= \frac{1}{2\sigma^2} \cdot \frac{r}{2}[\sum_{j=1}^{t-1} a_j^2 - \sum_{j=1}^{t-2} a_j a_{j+1}] \\ &= \frac{1}{2\sigma^2} \cdot \frac{b}{t}[\sum_{j=1}^{t-1} a_j^2 - \sum_{j=1}^{t-2} a_j a_{j+1}]. \end{aligned}$$

under H_a. Here, $\nu_1 = t-1$ and $N - rx = 2b - (b+t-1) = b-t+1$. For the special case $\{a_j\} = a1_{t-1}$, we have under H_a,

$$\begin{aligned}\delta &= \frac{1}{2\sigma^2} \cdot \frac{b}{t}[\sum_{j=1}^{t-1} a_j^2 - \sum_{j=1}^{t-2} a_j a_{j+1}] \\ &= \frac{1}{2\sigma^2} \cdot \frac{a^2 b}{t}.\end{aligned}$$

3.2.4 Illustration of The Three Designs

In summary, to test $H_0 : K'T = 0$ vs $H_a : K'T = \{a_j\}$, the test statistic $\frac{Q/\nu_1}{SSE/(N-rx)}$ has a central F distribution with parameters ν_1, and $\nu_2 = (N-rx)$ under H_0, and has a non-central F distribution with parameters ν_1, ν_2, and non-centrality parameter δ under H_a. Table 4 summarizes these parameters for the three designs. To have the three designs comparable, the pair (t, b) has to satisfy the conditions that both b/t and $2b/t(t-1)$ are integers.

We consider a design better if it is easier to reject the null hypothesis and detect the differentially expressed genes, i.e. it has a better power function. If the degrees of freedom are the same, a design with a larger non-centrality parameter indicates that the test statistic under the alternative is more skewed; thus it is more likely to reject the null hypothesis and so is a better design.

Define the non-centrality parameters of the BIB design, the reference design, and the Loop design to be δ_1, δ_2, and δ_3. Table 4 shows that $\delta_1 > \delta_2$ for any t. Since the degrees of freedom (ν_1 and ν_2) are almost the same, we can conclude that with the same number of slides and the same number of treatments, the BIB design is better than the reference design in terms of detecting a single differentially expressed gene.

The comparison between the BIB design and the loop design depends on the number of treatments and the order of the treatments in the loop. For $t = 2$ or 3, they are the same. For $t > 3$, $2\sigma^2(\delta_1 - \delta_3) = \{a_j\}'[C_{BIB} - C_{loop}]\{a_j\}$. The matrix $C_{BIB} - C_{loop}$ is not positive definite, hence we cannot determine which one of the BIB or the loop designs is better. For example, for $t = 4$, the size of $\{a_j\}$ is 3×1, if $\{a_j\}' = (1\ 2\ 3)$, then $\delta_1 = \frac{5}{3}b > \delta_3 = \frac{3}{2}b$; if $\{a_j\}' = (1\ 3\ 2)$, $\delta_1 = \frac{5}{3}b > \delta_3 = \frac{5}{4}b$; but if $\{a_j\}' = (2\ 1\ 3)$, $\delta_1 = \frac{5}{3}b < \delta_3 = \frac{9}{4}b$. In general, if the contrasts of each treatment to

the last treatment $T_j - T_t = a_j$ are not all the same (i.e., $a_j \neq a_{j'}$ for some $j \neq j'$), the loop design is better than the BIB design provided that the treatments are not clustered; say the effects are distributed evenly in the loop. That is, big effects are not clustered together and small effects are not clustered together either. For example, if the treatments are ordered in the loop to have effects such as $\{a\} = (1\ 4\ 1\ 4\ 1\ 4\ 1\ 4)$, the loop design will be better than the BIB design ($\delta_1 = 2.94b < \delta_3 = 4.44b$); but if they are ordered so that $\{a\} = (1\ 1\ 1\ 1\ 4\ 4\ 4\ 4)$, the loop design will be worse than the BIB design ($\delta_1 = 2.94b > \delta_3 = 1.44b$).

design	ν_1	ν_2	$\delta : K'T = \{a_j\}$	$\delta : K'T = a1_{t-1}$
BIB	$t-1$	$b-t+1$	$\delta_1 = \frac{1}{2\sigma^2} \cdot \frac{b}{t(t-1)}[t\sum_{j=1}^{t-1} a_j^2 - (\sum_{j=1}^{t-1} a_j)^2]$	$\frac{1}{2\sigma^2} \cdot \frac{a^2 b}{t}$
Reference	$t-1$	$b-t$	$\delta_2 = \frac{1}{2\sigma^2} \cdot \frac{b}{2t^2}[t\sum_{j=1}^{t-1} a_j^2 - (\sum_{j=1}^{t-1} a_j)^2]$	$\frac{1}{2\sigma^2} \cdot \frac{a^2 b(t-1)}{2t^2}$
Loop	$t-1$	$b-t+1$	$\delta_3 = \frac{1}{2\sigma^2} \cdot \frac{b}{t}[\sum_{j=1}^{t-1} a_j^2 - \sum_{j=1}^{t-2} a_j a_{j+1}]$	$\frac{1}{2\sigma^2} \cdot \frac{a^2 b}{t}$

Table 4: The parameters in the F-statistics for testing $H_a : K'T = \{a_j\}$ of three designs. Here, both b/t and $2b/(t(t-1))$ should be integers. $K' = (I_{t-1} \ -1_{t-1})$ in BIB and Loop designs, and $K' = (I_{t-1} \ -1_{t-1}, \ 0)$ in the reference design, where the 0 is the coefficient for the reference condition.

The same situation occurs when we compare the loop design to the reference design. When $t = 2$ or $t = 3$, the loop design is better since it is the same as the BIB design, which is better than the reference design. For $t > 3$, let us consider $2\sigma^2(\delta_3 - \delta_2) = \{a_j\}'[C_{loop} - U]\{a_j\}$. When $t = 4$, the matrix

$$[C_{loop} - U] = \begin{pmatrix} 5 & 3 & 1 \\ 3 & 5 & 3 \\ 1 & 3 & 5 \end{pmatrix},$$

and when $t = 5$,

$$[C_{loop} - U] = \begin{pmatrix} 6 & 4 & 1 & 1 \\ 4 & 6 & 4 & 1 \\ 1 & 4 & 6 & 4 \\ 1 & 1 & 4 & 6 \end{pmatrix}.$$

These two matrices are positive definite, so that $\delta_3 - \delta_2 > 0$ for any $\{a_j\}$; thus the loop design is better. However, when $t > 5$, the matrix $[C_{loop} - U]$ is not positive definite anymore. The loop design is sometimes better than the reference design but sometimes is worse. For example, if we order the treatments in the loop to have effects such as $\{a\} = (1\ 4\ 1\ 4\ 1\ 4\ 1\ 4)$, the loop will be better than the reference design ($\delta_3 = 4.44b > \delta_2 = 1.31b$), but if we order them so that $\{a\} = (1\ 4\ 4\ 4\ 4\ 1\ 1\ 1)$, the loop will be worse than the reference design ($\delta_3 = 1.11b < \delta_2 = 1.31b$).

For the special case $\{a_j\} = a1_{t-1}$, table 4 shows that the non-centrality parameter δ of the BIB design is the same as the δ of the Loop design, and is bigger than the δ of the reference design. Therefore, with the same number of slides and same number of treatments, if all $t - 1$ treatments have the same effect and differ by a constant from treatment t, the BIB design is the same as the loop design, and better than the reference design, in terms of detecting a single differentially expressed gene.

In conclusion, if there is no knowledge about the size of the treatment effects before the experiment, it can be assumed that only one treatment has a non-zero effect so it has a constant difference from all the other treatments. Alternatively, assume that there is a control group, and all other treatment effects differ by a constant from the control group. In this case, the test $H_a : K'T = a1_{t-1}$ will be used. Both the BIB design and the loop design are better than the reference design. The BIB design may be a little more complicated than the loop design in practice since each pair of treatments needs to be hybridized together. However, if in fact, the treatment effects differ a lot ($H_a : K'T = \{a_i\}$, $a_i - a_j$ are far from 0 for some i and j), the treatments with large effects might be clustered in the loop design. Then the BIB design is better than the loop design. If some knowledge about the size of the treatment effects is available and the sizes are different(e.g, based on literature review or previous experiments), then the treatments can be carefully located in the

	replicate r			replicate r	
(t,b)	BIB/Loop	Reference	(t,b)	BIB/Loop	Reference
(2,4)	4	2	(6,30)	10	5
(2,6)	6	3	(6,60)	20	10
(2,8)	8	4	(6,90)	30	15
(3,6)	4	2	(8,56)	14	7
(3,9)	6	3	(8,112)	28	14
(3,12)	8	4	(8,168)	42	21
(4,12)	6	3	(10,90)	18	9
(4, 24)	12	6	(10,180)	36	18
(4, 36)	18	9	(10,270)	54	27

Table 5: Illustration of replicates per treatment for (t,b) pairs in different designs. Simulations will be generated on these (t,b) pair in following chapters to demonstrate methods discussed there.

loop evenly. In this case the loop design might be better than the other two designs.

We will use the special alternative $H_a : K'T = a1_{t-1}$ as an example in the following chapters to illustrate the methods discussed there. Table 5 lists the (t,b) pairs we will use and the corresponding number of times (replicates) each treatment appears in the different designs. Table 6 illustrates the non-centrality parameter δ for each (t, b) combination. Figure 7 shows the values of δ with $t = 3$, $b = 6$ *or* 12, $\sigma = 1$ and $a \in [0, 4]$ for illustration.

3.3 Density of The P-values

We will discuss multiple comparison methods in the following chapters and the distributions of the test statistics $F(H)$ and their p-values under H_0 and H_a are the keys to developing sample size estimates there. We will describe the distributions here.

It is easy to show that under $H_0 : K'T = 0$ for a specific gene g, the p-value p of

	The non-centrality parameter δ							
	a=0.5		a=1		a=2		a=4	
(t,b)	BIB/Loop	Ref	BIB/Loop	Ref	BIB/Loop	Ref	BIB/Loop	Ref
(2,4)	0.25	0.06	1	0.25	4	1	16	4
(2,6)	0.375	0.09	1.5	0.375	6	1.5	24	6
(2,8)	0.5	0.125	2	0.5	8	2	32	8
(3,6)	0.25	0.08	1	0.33	4	1.3	16	5.33
(3,9)	0.375	0.125	1.5	0.5	6	2	24	8
(3,12)	0.5	0.17	2	0.67	8	2.67	32	10.67
(4,12)	0.375	0.14	1.5	0.56	6	2.25	24	9
(4,24)	0.75	0.28	3	1.125	12	4.5	48	18
(4,36)	1.125	0.42	4.5	1.69	18	6.75	72	27

Table 6: Illustration of the non-centrality parameter δ for different (t,b) pairs under $K'T = a1_{t-1}$. Here $\sigma = 1$ is assumed.

the F test statistic is uniformly distributed on [0,1]:

$$\begin{aligned} P[p \leq x] &= P[(1 - F_0(F(H))) \leq x] \\ &= 1 - F_0(F_0^{-1}(1-x)) \\ &= x \end{aligned}$$

where F_0 is the distribution function of central F, $x \in [0,1]$, and $p = 1 - F_0(F(H))$.

Under the alternative, $H_a : K'T = \{a_j\}$, where $\{a_j\}$ is a non-zero vector, the distribution of the p-value of the F-test statistic is

$$\begin{aligned} D(x) &= P[p \leq x] = P[1 - F_0(F(H)) \leq x] \\ &= \int_{F_0^{-1}(1-x)}^{\infty} f(t,\delta)dt, \end{aligned}$$

for $x \in [0,1]$. Thus, $D(x)$ is related to δ so we will rewrite it as $D_\delta(x)$. In this calculation, $F(t,\delta)$ and $f(t,\delta)$ are the distribution function and density function respectively

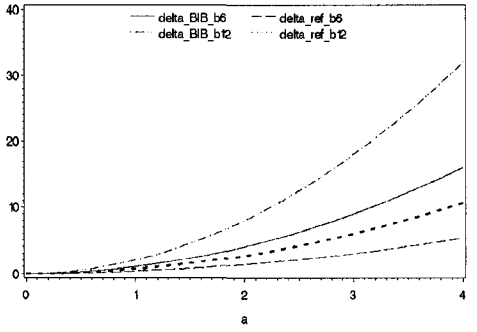

Figure 7: Illustration of the non-centrality parameter δ for t=3, b=6 and 12, and $\sigma = 1$, under $K'T = a1_{t-1}$. Here a is chosen in $[0, 4]$.

of a non-central F distribution with non-centrality parameter $\delta = \{a_j\}'(S'GS)^{-1}\{a_j\}$ (see previous sections for calculation). Under the alternative hypothesis specified above, the distribution of $F(H)$ is actually $F(\cdot, \delta)$.

We can calculate the density function $d_\delta(x)$ of p-values under the alternative hypothesis:

$$\begin{aligned} d_\delta(x) &= \frac{d}{dx} D_\delta(x) \\ &= -(f(F_0^{-1}(1-x)), \delta) \cdot \frac{dF_0^{-1}(1-x)}{dx} \\ &= (f(F_0^{-1}(1-x)), \delta)/(f_0(F_0^{-1}(1-x))) \\ &= e^{-\frac{\delta}{2}} \sum_{j=0}^{\infty} \left[\frac{\frac{1}{2}\delta\nu_1 F_0^{-1}(1-x)}{\nu_2 + \nu_1 F_0^{-1}(1-x)} \right]^j \cdot \frac{(\nu_1+\nu_2)(\nu_1+\nu_2+2)\cdots(\nu_1+\nu_2+2(j-1))}{j!\ \nu_1(\nu_1+2)\cdots(\nu_1+2(j-1))}, \end{aligned}$$

for $x \in [0, 1]$. Here ν_1 and ν_2 are the degrees of freedom of the numerator and denominator in the non-central F test statistics. The last equation is obtained directly from the non-central F distribution formula of [18, page 191]. As the density of a continuous random variable, $d_\delta(\cdot)$ certainly exists on $(0, 1)$, and as a differentiable monotone transform of an F distributed random variable (itself with a

smooth density), it will also be smooth. When x increases, $F_0^{-1}(1-x)$ decreases and $\nu_1 F_0^{-1}(1-x)/(\nu_2+\nu_1 F_0^{-1}(1-x)) = \nu_1/(\nu_2/F_0^{-1}(1-x)+\nu_1)$ decreases. Therefore, $d_\delta(x)$ is monotonically decreasing on $x \in (0,1)$ for fixed δ, ν_1, and ν_2. Figure 8 illustrates the plot of $d_\delta(x)$ for $\nu_1 = 2$, $\nu_2 = 3$, and different δs. We can also find an upper bound of $d_\delta(x)$: since

$$\begin{aligned} A^j &= \frac{(\nu_1+\nu_2)(\nu_1+\nu_2+2)\cdots(\nu_1+\nu_2+2(j-1))}{\nu_1(\nu_1+2)\cdots(\nu_1+2(j-1))} \\ &= \frac{\nu_1+\nu_2}{\nu_1}\cdot\frac{\nu_1+\nu_2+2}{\nu_1+2}\cdots\frac{\nu_1+\nu_2+2(j-1)}{\nu_1+2(j-1)} \\ &= (1+\frac{\nu_2}{\nu_1})\cdot(1+\frac{\nu_2}{\nu_1+2})\cdots(1+\frac{\nu_2}{\nu_1+2(j-1)}) \\ &\leq (1+\frac{\nu_2}{\nu_1})^j, \end{aligned}$$

we have

$$\begin{aligned} d_\delta(x) &= e^{-\frac{\delta}{2}}\sum_{j=0}^{\infty}\left[\frac{\frac{1}{2}\delta\nu_1 F_0^{-1}(1-x)}{\nu_2+\nu_1 F_0^{-1}(1-x)}\right]^j\cdot A^j/j! \\ &\leq e^{-\frac{\delta}{2}}\sum_{j=0}^{\infty}\left[\frac{\frac{1}{2}\delta\nu_1 F_0^{-1}(1-x)}{\nu_1 F_0^{-1}(1-x)}\right]^j\cdot A^j/j! \\ &= e^{-\frac{\delta}{2}}\sum_{j=0}^{\infty}\left[\frac{1}{2}\delta\right]^j\cdot A^j/j! \\ &\leq e^{-\frac{\delta}{2}}\sum_{j=0}^{\infty}\left[\frac{1}{2}\delta\right]^j\cdot(1+\frac{\nu_2}{\nu_1})^j/j! \\ &= e^{-\frac{\delta}{2}}\cdot e^{\frac{\delta}{2}(1+\frac{\nu_2}{\nu_1})} \\ &= e^{\frac{\delta}{2}\frac{\nu_2}{\nu_1}}. \end{aligned}$$

Thus, for fixed δ, ν_1, and ν_2, $d_\delta(x)$ can be bounded above by a constant for all x in $(0,1)$.

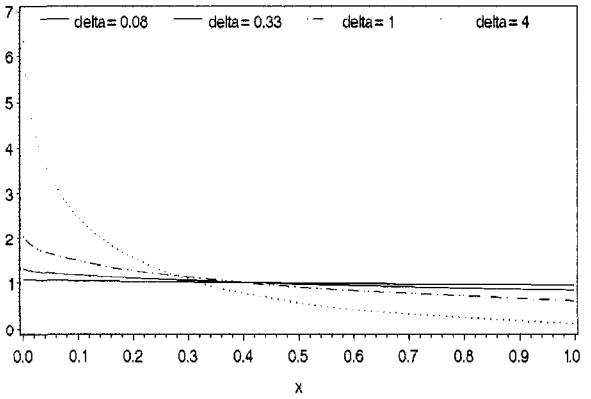

Figure 8: The density function $d_\delta(x)$ of p-values under alternative hypothesis $K'T = a1_{t-1}$, for $t = 3$, $b = 6$. That is, $\nu_1 = 2$, and $\nu_2 = 3$. The δ used here is chosen from the previous table for illustration.

The asymptotic properties of extreme p-values will be dependent on the character of the left tail of the densities $d_\delta(x)$, in particular the rate at which $D_\delta(x)$ approaches 0 as x decreases to 0. Since $d_\delta(x)$ is monotone and bounded, we can conclude that $d_\delta(0+)$ exists. However, due to the limitation of software, we cannot calculate the exact value of $d_\delta(0+)$ since the number x we can use for SAS should be bigger than 0.000001. Consequently the exact limiting behaviour of minimal p-values is not accessible. However, we know when x decreases to 0, $F_0^{-1}(1-x)$ increases to ∞, and $d_\delta(x)$ is a function of $F_0^{-1}(1-x)$. For large enough $F_0^{-1}(1-x)$, say 10000 or 100000, $d_\delta(x)$ converges very quickly. Therefore, we could use the value of $d_\delta(x)$ with a large $F_0^{-1}(1-x)$ as an approximation of $d_\delta(0)$. Tables 7 and 8 illustrate $d_\delta(x)$ with $F_0^{-1}(1-x) = 10000$ and 100000 for different t, b, a combinations and designs. It shows that for moderate b and a, which will lead to a moderate δ, the value of $d_\delta(x)$ with $F_0^{-1}(1-x) = 10000$ is very close to the value with $F_0^{-1}(1-x) = 100000$. Thus, the value of $d_\delta(x)$ at $F_0^{-1}(1-x) = 10000$ could be used as a very good approximation for $d_\delta(0)$. The difference becomes bigger for larger b or a (larger δ). In this situation, a larger number than $F_0^{-1}(1-x)$ could be used to calculate $d_\delta(x)$ in case a good approximation of $d_\delta(0)$ is needed.

In conclusion, this chapter has considered several standard experimental designs

				BIB design			Reference design		
t	b	a	$F_0^{-1}(1-x)$	δ	$d_\delta(x)$	g	δ	$d_\delta(x)$	g
2	4	1	10000	1	2.62	1.02	0.25	1.25	1.00
			100000	1	2.62	1.02	0.25	1.25	1.00
2	6	1	10000	1.5	6.22	1.05	0.375	1.80	1.01
			100000	1.5	6.22	1.05	0.375	1.80	1.01
2	8	1	10000	2	15.01	1.14	0.5	2.76	1.02
			100000	2	15.04	1.14	0.5	2.76	1.02
2	10	1	10000	2.5	36.33	1.35	0.625	4.34	1.03
			100000	2.5	36.42	1.35	0.625	4.35	1.03
2	20	1	10000	5	3009	31	1.25	45.08	1.44
			100000	5	3042	31	1.25	45.28	1.44
2	30	1	10000	7.5	247816	2479	1.875	470.16	5.69
			100000	7.5	254041	2541	1.875	475.09	5.74
2	4	2	10000	4	8.77	1.08	1	2.00	1.01
			100000	4	8.78	1.08	1	2.00	1.01
2	6	2	10000	6	42.50	1.41	1.5	4.75	1.04
			100000	6	42.59	1.42	1.5	4.75	1.04
2	8	2	10000	8	205.95	3.05	2	11.52	1.11
			100000	8	206.77	3.06	2	11.53	1.11
4	12	1	10000	1.5	4.65	1.04	0.563	1.88	1.01
			100000	1.5	4.66	1.04	0.563	1.88	1.01
4	24	1	10000	3	110.12	2.09	1.125	10.56	1.10
			100000	3	110.43	2.09	1.125	10.58	1.10
4	36	1	10000	4.5	3539	36	1.688	81.07	1.80
			100000	4.5	3563	37	1.688	81.36	1.80
4	12	2	10000	6	45.88	1.45	2.25	6.48	1.05
			100000	6	45.95	1.45	2.25	6.49	1.05
4	24	2	10000	12	28566	287	4.5	330.85	4.30
			100000	12	28769	289	4.5	331.97	4.31
4	36	2	10000	18	24014244	240143	6.75	23169	233
			100000	18	24426698	244268	6.75	23367	235

Table 7: $d_\delta(x)$ with $F_0^{-1}(1-x) = 10000$ and 100000 for different t, b, a combinations and designs. Column g will be explained and used later.

				BIB design			Reference design		
t	b	a	$F_0^{-1}(1-x)$	δ	$d_\delta(x)$	g	δ	$d_\delta(x)$	g
6	30	1	10000	2.5	41.95	1.41	1.042	6.52	1.06
			100000	2.5	42.02	1.41	1.042	6.53	1.06
6	60	1	10000	5	34036	341	2.083	357.05	4.56
			100000	5	34335	344	2.083	358.79	4.58
6	90	1	10000	7.5	4.8E+7	478563	3.125	33253	334
			100000	7.5	4.9E+7	488786	3.125	33660	338
6	30	2	10000	10	8346	84	4.167	167.85	2.67
			100000	10	8385	85	4.167	168.24	2.67
6	60	2	10000	20	7.1E+9	7.1E+7	8.333	1.1E+6	11206
			100000	20	7.2E+9	7.2E+7	8.333	1.1E+6	11338
6	90	2	10000	30	1.1E+16	1.1E+14	12.5	1.3E+10	1.3E+8
			100000	30	1.1E+16	1.1E+14	12.5	1.4E+10	1.4E+8
8	56	1	10000	3.5	791.19	8.90	1.531	36.99	1.36
			100000	3.5	794.34	8.93	1.531	37.07	1.36
8	112	1	10000	7	5.4E+7	544146	3.063	37629	377
			100000	7	5.5E+7	554658	3.063	38056	382
8	168	1	10000	10.5	8.3E+12	8.3E+10	4.594	8.2E+7	824649
			100000	10.5	8.7E+12	8.7E+10	4.594	8.5E+7	847745
8	56	2	10000	14	7.9E+6	78944	6.125	13522	136
			100000	14	8.0E+6	79745	6.125	13599	137
8	112	2	10000	28	5.6E+16	5.6E+14	12.25	4.4E+10	4.4E+8
			100000	28	5.9E+16	5.9E+14	12.25	4.6E+10	4.6E+8
8	168	2	10000	42	9.1E+26	9.1E+24	18.375	3.3E+17	3.3E+15
			100000	42	1.0E+27	1.0E+25	18.375	3.5E+17	3.5E+15
10	90	1	10000	4.5	28640	287	2.025	324.61	4.24
			100000	4.5	28853	290	2.025	325.99	4.25
10	180	1	10000	9	3.6E+11	3.6E+9	4.05	1.0E+7	106583
			100000	9	3.7E+11	3.7E+9	4.05	1.1E+7	108861

Table 8: Continue of table 7.

for comparing treatments in the context of microarray experiments, established non-centrality parameters under a standard alternative, and related that non-centrality parameter to an upper bounds and an approximation on the rate of growth of $D_\delta(x)$ at 0.

Chapter 4

Power Based on False Discovery Rate

For high dimensional multiple tests involved in microarray experiments, the traditional multiple test corrections such as the Bonferroni or Sidak techniques become too conservative. The thresholds in these techniques are too small and many interesting genes worthy of further attention will not be detected. Other less conservative techniques such as the false discovery rate have been developed in recent papers to overcome this high multiplicity problem. These methods generally belong either to permutation based methods or model based methods. Allison et al. [2, Chapter 5] summarized these techniques.

Among the methods proposed in recent papers on microarray analysis, usually test statistics are computed for a single gene at a time in order to compare the differential expression of genes under different conditions. Then p-values are calculated by using either permutation or bootstrap procedures or by using an assumed model for the test statistics. The distribution of p-values could then be modeled by some density functions if possible. It is easy to show that under the null hypothesis the p-values based on a continuous test statistic have uniform distributions. Under the alternative, Hung et al. [16] illustrated that the distribution of p-values is a function of both sample size of the experiment and the true value of the parameter. They derived the exact distribution of p-values of t test statistics. Allison et al. [1] and Pounds and

Morries [27] modeled a distribution of p-values using a mixture of a uniform and beta distributions.

We derived the exact distribution of p-values of F test statistics under alternative hypotheses in Chapter 3. In this chapter, we will address the false discovery rate method and focus on the number of genes that are detected as differentially expressed. We review the definition of the FDR method in section 1 and list our questions in section 2. In section 3, we calculate lower bounds for the probability that at least k genes are detected using the FDR method, and for the probability that given m_1 truly differentially expressed genes, j of them are detected. Examples of how to apply our results to design problems are given. Section 4 gives numerical assessments of the lower bounds. In section 5, a Poisson approximation is developed as an alternate lower bound for the probability that at least k genes are detected as significant. Simulation results show that the Poisson approximation for the lower bound is good only for moderate mean values.

A fundamental assumption of this chapter is that the p-values of tests for separate genes are statistically independent. We know that the biological dependencies of differential expression levels among genes are present in every microarray study. Although in data analysis, applying resampling methods such as the bootstrap or permutation procedure allow us to preserve the dependence structure between the genes, it is virtually impossible to include these unknown dependencies at the design stage. Since our focus is on experimental design and not data analysis, we assume statistical independence for simplicity.

4.1 False Discovery Rate

The false discovery rate (FDR) was first introduced by Benjamini and Hochberg [3] as a procedure for multiple-hypothesis testing. The procedure was described in the first chapter, which we now summarize for convenience. We adopt a random mechanism for the number of genes coming from the alternative hypotheses. This random setting is as in Efron [12], despite the fact that we do not pursue a Bayesian approach.

Assume that among m simultaneous hypotheses H_1^0, H_2^0,$\cdots$, H_m^0, M_0 of them are

test hypothesis	test declaration Not-Significant	Significant	total
True	U	V	m_0
False	T	S	m_1
total	m-R	R	m

Table 9: Multiple Testing Framework.

"true null hypotheses" and m_0 is the value of M_0 in a given experiment (see Table 9). Here, $M_0 \sim Binom(m, \lambda)$, with λ assumed to be a known constant. It would vary widely from one experiment to another. Since usually we don't expect to have a large number of differentially expressed genes, it is reasonable to assume $\lambda = 0.99$ for the purpose of illustration. Using the notation in Table 9, the false discovery rate is $Q = V/R$ - the proportion of the rejected null hypotheses which are erroneously rejected. For the m independent tests denote the test statistics by $X_1, X_2, \cdots, X_m$, with corresponding p-values $p_1, p_2, \cdots, p_m$. Denote the ordered observed p-values as $p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(m)}$. The Benjamini and Hochberg procedure is: for $0 < q^* < 1$, let k be the largest i for which

$$p_{(i)} \leq \frac{i}{m} q^*, \tag{4.1.1}$$

then reject all of the corresponding null hypotheses $H^0_{(1)}, \cdots, H^0_{(k)}$. When the test statistics $X_1, X_2, \cdots, X_m$ are independent, among the k rejected tests the expectation of the false discovery rate $E(Q)$ is less than or equal to q^*.

4.2 Questions To Be Studied

The FDR method is less conservative than the Bonferroni or Sidak methods and is used in microarray data to overcome the high dimension testing problem. It can control the FDR at a given level q^*, that is $E(Q) \leq q^*$. But it is of interest to know first how likely it is to reject more than k hypotheses using this method for a given test, $P[R > k]$. If the probability of rejecting a moderate number of null hypotheses

is very small or very large, it would be helpful to know this fact before applying the method.

Of course it is always of interest to know the power of a test. The power of a single test is defined as the probability of rejecting the null hypothesis when it is false. For several simultaneous tests, the direct generalization of power will be the probability $P(\textit{reject at least } j \textit{ of them when there are } m_1 \textit{ false hypotheses})$. That is, $P(S \geq j|m_1)$.

Moreover, since both S and m_1 are unknown while R is observable, it will still be useful to predict the actual number of true positives for a given number of rejections. That is, $P[S = j|R = k]$. For example, $S = 2$ will be much better in a design with $R = 10$ than with $R = 100$. This leads to interest in the probability of rejecting exactly k hypotheses, $P[R = k]$. We will use the p-values generated from the F-test statistics in Chapter 3 and study these concerns:

a) Given the number of slides to be used, how big is the probability that k genes are considered as differentially expressed if the FDR procedure is applied, i.e. $P(R = k)$?

b) Given R apparently differentially expressed genes, what is the probability that the number of truly differentially expressed genes (i.e. true positive) exceeds a certain cutoff, say 2, 5, or 10? Given $R = k$, this probability can be expressed as $P(S \geq j|R = k)$.

   The bigger this probability, the better the test, and $P(S = k|R = k) = 1$ would be the ideal case.

c) If the alternative distribution is very skewed, the possible number of rejections k could have a large range and thus it would be more useful to consider $P[R > k]$.

d) Given that m_1 genes are true differentially expressed, what is the probability $P(S > j|m_1)$ that the number of true positives exceeds a cutoff?

We will establish an upper bound for $P[R = k]$, an approximation for $P[S \geq j|R = k]$, and lower bounds for $P(R > k)$ and $P(S > j|m_1)$ for the case of $M_1 \sim Binom(m, 1 - \lambda)$ genes from the alternative.

4.3 Bounds and Applications

4.3.1 Bounds

Assuming that the test statistics of the truly differentially expressed genes have a common non-central F distribution with non-centrality parameter δ, the following theorem gives an upper bound for $P[R = k]$ and lower bounds for $P(R > k)$ and $P(S > j|m_1)$.

Theorem 4.3.1 *Assume all single gene tests are independent, and that*

a. $M_0 \sim Binom(m, \lambda)$, *for some* $0 < \lambda < 1$, *and*

b. *the test statistics of the truly differentially expressed genes are determined by non-central F distributions with a fixed common non-centrality parameter δ, let $D(x)$, $d_\delta(x)$ be the distribution and density functions of such p-values, (the subscript δ in $D(x)$ is omitted for notational simplicity)*

Let $M_1 = m - M_0$. For any integer l between k and m, that is, $0 \le k < l \le m$, we have

i.

$$\begin{aligned} P(R=k) \;\le\; & min[1, \sum_{S1} \frac{m!}{(m-k-\sum r_i)!k!\prod_i r_i!}[\lambda\frac{k}{m}q^* + (1-\lambda)D(\frac{k}{m}q^*)]^k \\ & \cdot \prod_{i=1}^{l-k-1}[\lambda\frac{q^*}{m} + (1-\lambda)(D(\frac{k+1+i}{m}q^*) - D(\frac{k+i}{m}q^*))]^{r_i} \\ & \cdot[\lambda(1-\frac{l}{m}q^*) + (1-\lambda)(1-D(\frac{l}{m}q^*))]^{m-k-\sum r_i}]; \end{aligned}$$

where $S1 = \{0 \le r_1 \le 1, 0 \le r_2 \le 2-r_1, \cdots, 0 \le r_{l-k-1} \le l-k-1-\sum_1^{l-k-2} r_i\}$.

ii.

$$P[S = j \mid R = k] \approx \binom{k}{j} w^j(1-w)^{k-j};$$

where $w = \frac{(1-\lambda)d_\delta(0)}{\lambda+(1-\lambda)d_\delta(0)}$;

iii.

$$P[R > k] > 1 - \sum_{S2} \frac{m!}{(m - \sum r_i)! \prod_i r_i!} [\lambda \frac{k+1}{m} q^* + (1-\lambda) D(\frac{k+1}{m} q^*)]^{r_1}$$
$$\cdot \prod_{i=2}^{l-k} [\lambda \frac{q^*}{m} + (1-\lambda)(D(\frac{k+i}{m} q^*) - D(\frac{k+i-1}{m} q^*))]^{r_i}$$
$$\cdot [\lambda(1 - \frac{l}{m} q^*) + (1-\lambda)(1 - D(\frac{l}{m} q^*))]^{m - \sum r_i}$$

where $S2 = \{0 \le r_1 \le k, 0 \le r_2 \le k+1-r_1, \cdots, 0 \le r_{l-k} \le l - \sum_1^{l-k-1} r_i\}$.

iv. Given $M_1 = m_1$, *for* $j < l \le m_1$,

$$P[S > j \mid M_1 = m_1] > 1 - \sum_{S3} \frac{m_1!}{\prod_i r_i! (m_1 - \sum r_i)!} [D(\frac{j+1}{m} q^*)]^{r_1}$$
$$\cdot \prod_{i=2}^{l-j} [D(\frac{j+i}{m} q^*) - D(\frac{j+i-1}{m} q^*)]^{r_i} \cdot [1 - D(\frac{l}{m} q^*)]^{m_1 - \sum r_i}$$

where $S3 = \{0 \le r_1 \le j, 0 \le r_2 \le j+1-r_1, \cdots, 0 \le r_{l-j} \le l - 1 - \sum r_i\}$.

Proof.

i. Define $N(x, y)$ to be the number of p-values such that $x < p \le y$. For $k < l \le m$,

$$\begin{aligned} P[R = k] &= P[p_{(k)} \le \frac{k}{m} q^*, p_{(k+1)} > \frac{k+1}{m} q^*, \cdots, p_{(m)} > \frac{m}{m} q^*] \\ &\le P[p_{(k)} \le \frac{k}{m} q^*, p_{(k+1)} > \frac{k+1}{m} q^*, \cdots, p_{(l)} > \frac{l}{m} q^*] \\ &= P[N(0, \frac{k}{m} q^*) = k, N(\frac{k}{m} q^*, \frac{k+1}{m} q^*) = 0, N(\frac{k+1}{m} q^*, \frac{k+2}{m} q^*) \le 1, \\ &\quad N(\frac{k+1}{m} q^*, \frac{k+3}{m} q^*) \le 2, \cdots, N(\frac{k+1}{m} q^*, \frac{l}{m} q^*) \le l - k - 1] \\ &= \sum_{S1} P[N(0, \frac{k}{m} q^*) = k, N(\frac{k}{m} q^*, \frac{k+1}{m} q^*) = 0, N(\frac{k+1}{m} q^*, \frac{k+2}{m} q^*) = r_1, \\ &\quad N(\frac{k+2}{m} q^*, \frac{k+3}{m} q^*) = r_2, \cdots, N(\frac{l-1}{m} q^*, \frac{l}{m} q^*) = r_{l-k-1}] \end{aligned}$$

where $S1 = \{0 \le r_1 \le 1, 0 \le r_2 \le 2 - r_1, \cdots, 0 \le r_{l-k-1} \le l - k - 1 - \sum_1^{l-k-2} r_i\}$.

Since the p-values either come from $U(0,1)$ (with probability λ) or from $D(x)$ (with probability $1-\lambda$), the m p-values can be considered as if they come from a mixed distribution with distribution function $F(x) = \lambda x + (1-\lambda)D(x)$. Because all p-values are independent, the numbers of p-values falling into segments of [0,1] have a multinomial distribution. Thus, the probability of having exact k observations in $(0, \frac{k}{m}q^*)$ is $F^k(\frac{k}{m}q^*)$, the probability of having exact r_1 observations in $(\frac{k+1}{m}q^*, \frac{k+2}{m}q^*)$ is $[F(\frac{k+2}{m}q^*) - F(\frac{k+1}{m}q^*)]^{r_1}$, and so on. Plugging these probabilities into the last expression will give the result.

ii. Given $R = k$, the first k smallest p-values are all smaller than $\frac{k}{m}q^*$ and are either from the alternative or null hypotheses. The probability that p_j comes from the alternative can be approximated as follows:

$$\begin{aligned}
& P[p_j \text{ is from "}H_1\text{" and } p_j \le \frac{k}{m}q^*]/P[p_j \le \frac{k}{m}q^*] \\
= \; & P[p_j \le \frac{k}{m}q^* | p_j \text{ is from "}H_1\text{"}]P[p_j \text{ is from "}H_1\text{"}]/P[p_j \le \frac{k}{m}q^*] \\
= \; & D(\frac{k}{m}q^*) \cdot (1-\lambda)/[D(\frac{k}{m}q^*) \cdot (1-\lambda) + \lambda\frac{k}{m}q^*] \\
\approx \; & \frac{k}{m}q^* d_\delta(0)(1-\lambda)/[\frac{k}{m}q^* \cdot (d_\delta(0)(1-\lambda) + \lambda)] \\
= \; & \frac{d_\delta(0)(1-\lambda)}{d_\delta(0)(1-\lambda)+\lambda} := w.
\end{aligned}$$

Therefore, we have $P[S = j \mid R = k] \approx \binom{k}{j} w^j (1-w)^{k-j}$, and $P[S \ge j | R = k] = 1 - \sum_{i=0}^{j-1} P[S = i | R = k]$.

iii. Since

$$\begin{aligned}
P[R \le k] &= P[\text{at most } k \textit{ hypotheses are rejected}] \\
&\le P[p_{(k+1)} > \frac{k+1}{m}q^*, p_{(k+2)} > \frac{k+2}{m}q^*, \cdots, p_{(l)} > \frac{l}{m}q^*] \\
&= P[N(0, \frac{k+1}{m}q^*) \le k, \cdots, N(0, \frac{l}{m}q^*) \le l-1]
\end{aligned}$$

Similar to part (i), we can obtain an upper bound for $P[R \leq k]$. Plugging it into $1 - P[R \leq k]$ will give the result.

iv. Let w_i be the p-values from the alternative and $w_{(i)}$ be the corresponding ordered p-values. Then given $M_1 = m_1$, we have

$$[S \leq j|m_1] \Rightarrow w_{(j+1)} > \frac{j+1}{m}q^*, \cdots, w_{(m_1)} > \frac{m_1}{m}q^*.$$

Define $N_2(x, y)$ be the number of p-values w_i such that $x < w_i \leq y$, then

$$\begin{aligned} P[S \leq j|m_1] &\leq P[w_{(j+1)} > \frac{j+1}{m}q^*, \cdots, w_{(m_1)} > \frac{m_1}{m}q^* \mid m_1] \\ &= P[N_2(0, \frac{j+1}{m}q^*) \leq j, \cdots, N_2(0, \frac{m_1}{m}q^*) \leq m_1 - 1] \\ &\leq \sum_{S3} P[N_2(0, \frac{j+1}{m}q^*) = r_1, N_2(\frac{j+1}{m}q^*, \frac{j+2}{m}q^*) = r_2, \cdots, \\ &\quad N_2(\frac{l-1}{m}q^*, \frac{l}{m}q^*) = r_{l-j}] \\ &= \sum_{S3} \frac{m_1!}{\prod r_i!(m_1 - \sum r_i)!}[D(\frac{j+1}{m}q^*)]^{r_1} \prod_{i=2}^{l-j}[D(\frac{j+i}{m}q^*) - \\ &\quad D(\frac{j+i-1}{m}q^*)]^{r_i} \cdot [1 - D(\frac{l}{m}q^*)]^{m_1 - \sum r_i} \end{aligned}$$

Plugging the last expression into $P[S > j|m_1] = 1 - P[S \leq j|m_1]$ will give a lower bound for $P[S > j|m_1]$. □

Although the bounds in items i, ii, and iv have cumbersome expressions, simulation results show that they are quite close to the probabilities they bound even with a small l such as $l = k+5$ for most (t, b, a) combinations, especially the lower bounds of $P[R > k]$. For such a small l, the number of terms in the summations are reasonable so that it doesn't take too long to calculate these bounds. Simulation results also indicate that the difference between bounds with $l = k + 5$ and $l = k + 10$ are very small. Likewise, we expect that the lower bounds for $P[R > k]$ and $P[S > j|m_1]$ with $l = k + 5$ should be close to the actual probabilities they bound in many microarray experiments. Consequently, they could be used in experimental designs to control $P[R > k]$ and the power $P[S > j|m_1]$.

The bounds could be modified to accommodate p-values from different alternative distributions although the expression will be more complicated. For example, for $t = 2$, in some experiments it is more reasonable to assume that 1% of differentially expressed genes are 2 fold expressed in the treatment group while another 0.5% of genes are 4 fold expressed in the treatment group. The density function of p-values of genes in each group could be calculated and the expression of the bounds could be adjusted accordingly. Similar bounds could be developed for other test statistics, given that the distribution function of the corresponding p-values under the alternative has a finite first derivative at point 0.

4.3.2 Applications

A few applications of the lower bounds are illustrated in this section. The lower bounds can be referenced to design experiments so that the sample size is large enough to obtain useful findings; they can be used to compare the efficiency of designs and they can be used to determine the appropriate sample size for a given power.

Estimating Sample Size for $P[R > k]$ and Comparison of Designs

The difference between the lower bounds of $P[R > k]$ for different designs will demonstrate the degree of gain/loss for each design in terms of the number of rejections, which would be a measure that a designer could use to select a design taking into consideration costs and other constraints. The lower bound of $P[R > k]$ for different numbers of slides will provide a reference for sample size selection.

Example 1. To test $H_0 : K'T = 0$ *vs* $H_a : K'T = a1_{t-1}$.

Table 10 lists different sample sizes (number of slides) and the corresponding lower bound with $l = k + 5$ of $P[R > k]$ - the probability of rejecting more than k hypotheses using the FDR method for $t = 2$, $a = 2$, and $\sigma = 1$. In the table, it is assumed that $M_0 \sim Binom(10000, 0.99)$ of the genes in the experiment are truly differentially expressed genes. $a = 2$ indicates that these differentially expressed genes are four fold expressed in one group (treatment) than in the other group (control). The lower

bound of the BIB design is much bigger than the lower bound of the Reference design when b increases, which indicates that the BIB design is more efficient than the reference design in terms of detecting differentially expressed genes, as we expected. However, the lower bound is still very small in the BIB design, which indicates that the chance of rejecting a large number of genes using the FDR method is still small. For instance, if 100 of 10000 genes are truly differentially expressed genes, the table suggests that to compare two conditions using the FDR method even with 10 replicates for each gene with the BIB design, the lower bound of detecting more than 1 gene is only 0.36. The probability becomes large with 15 replicates.

Example 2. To test an alternative with distinct treatment effects H_a : $K'T = a_{t-1}$.

Table 11 illustrates the lower bounds of $P[R > k]$ for testing $K'T = \{1, 2, 1\}$ and $K'T = \{2, 1, 1\}$, with $t = 4$. To test alternatives with distinct treatment effects such as $K'T = \{1, 2, 1\}$, one needs to first calculate the non-centrality parameter δ as in table 4 and use the corresponding $D(x)$ to calculate the lower bounds of $P[R > k]$. In the table, it is assumed that $M_1 \sim Binom(10000, 0.01)$ of the genes in the experiment are truly differentially expressed genes.

For treatments with different effects, the order of the treatments has no effect on BIB design while it does affect the efficiency of the loop design. The BIB is better than the loop design in the $K'T = \{1, 2, 1\}$ setting (loop1) and worse in the $K'T = \{2, 1, 1\}$ setting (loop2). The table shows that for 36 slides, the probability of rejecting more than 10 genes is 0.45 for the BIB design, is 0.02 for the loop design which orders treatments in the loop such that $K'T = \{1, 2, 1\}$, and is 0.79 for the loop design which orders treatments such that $K'T = \{2, 1, 1\}$.

Estimating Sample Size for $P[S > j|m_1]$

Sometimes the power of tests is of more interest and it is necessary to calculate the sample size needed to reject at least j false hypotheses, if the total number of false hypotheses m_1 could be roughly predicted. The lower bound of $P[S > j|m_1]$ can be tabulated and used for estimating the required sample size. Of course it could also

be used to compare efficiencies of different designs if a large number of true rejections is a goal.

Example 3. To test $H_0 : K'T = 0$ *vs* $H_a : K'T = a1_{t-1}$.

Table 12 illustrates the lower bound of $P[S > j|m_1]$ for the BIB design of $t = 2$, and $a = 1$ or 2, with $m_0 = 9900$ p-values from $U(0,1)$ and $m_1 = 100$ p-values from the alternative distribution $D(x)$. The table can be used for sample size estimation. For example, when $a = 1$, even with 20 slides, the probability of rejecting at least 1 false hypothesis is only 0.33. In the case of $a = 2$, if one wants to reject more than 10 false hypotheses, 15 slides will be enough, and if one wants to reject 50 false hypotheses, 20 slides will be needed.

4.4 Numerical Assessments

Simulations were performed and the probabilities $P[R = k]$, $P[R > k]$, and $P[S = j|m_1]$ calculated from the simulation results were compared to the bounds developed in Theorem 4.3.1. $P[S \geq j|R = k]$ is not compared since it is approximately a standard Binomial mass function.

The test $H_0 : K'T = 0$ *vs* $H_a : K'T = a1_{t-1}$ in Chapter 3 was employed in the simulation and the FDR method was applied to the p-values of these tests. Ten thousand simulations were performed for each (t, b, a) combination. The choices of t used are $t = 2, 3, 4, 6, 8$, and 10. For each t, the smallest three choices of b which are suitable for both the BIB and reference designs are chosen here. That is, the (t,b) choices used for simulations were among (2,4), (2,6), (2,8), (3,6), (3,9), (3,12), (4,12), (4, 24), (4, 36), (6, 30), (6, 60), (6,90), (8, 56), (8, 112), (8, 168), (10, 90), (10, 180), and (10, 270). For each (t,b) pair, three choices of a : 0.5, 1, and 2 were applied, which corresponded to $\sqrt{2}$, double, and four fold gene expression differences. Simulations on the BIB and reference designs are performed since the loop design has the same non-centrality parameter as the BIB design in testing $H_0 : K'T = 0$ *vs* $H_a : K'T = a1_{t-1}$, which would lead to the same distribution of p-values under the alternative.

It is assumed that among a total of 10000 genes, $M_0 \sim Binom(10000, 0.99)$ of them

		lower bound of $P[R > k]$ for $a = 2, t = 2$									
design	b	k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9	k=10
BIB	4	0.0214	0.0047	0.0011	0.0003	6.3E-05	1.5E-05	3.8E-06	9.4E-07	2.3E-07	5.9E-08
	6	0.0347	0.0095	0.0027	0.0008	0.0002	7.0E-05	2.1E-05	6.4E-06	2.0E-06	6.0E-07
	8	0.1014	0.0431	0.0188	0.0082	0.0036	0.0016	0.0007	0.0003	0.0001	5.9E-05
	10	0.3601	0.2468	0.1689	0.1148	0.0774	0.0516	0.0341	0.0223	0.0145	0.0093
	15	0.9984	0.9978	0.9970	0.9959	0.9945	0.9927	0.9904	0.9874	0.9838	0.9792
	20	1	1	1	1	1	1	1	1	1	1
Ref	4	0.0189	0.0039	0.0009	0.0002	4.4E-05	1.0E-05	2.4E-06	5.5E-07	1.3E-07	3.1E-08
	6	0.0199	0.0042	0.0009	0.0002	5.1E-05	1.2E-05	2.9E-06	6.8E-07	1.7E-07	4.0E-08
	8	0.0222	0.0050	0.0012	0.0003	6.8E-05	1.7E-05	4.2E-06	1.1E-06	2.7E-07	6.7E-08
	10	0.0269	0.0065	0.0017	0.0004	0.0001	3.0E-05	8.0E-06	2.2E-06	5.8E-07	1.6E-07
	15	0.0652	0.0223	0.0078	0.0028	0.0010	0.0003	0.0001	4.3E-05	1.5E-05	5.2E-06
	20	0.2040	0.1068	0.0556	0.0286	0.0145	0.0073	0.0036	0.0018	0.0009	0.0004
	30	0.8328	0.7581	0.6800	0.6006	0.5223	0.4471	0.3768	0.3128	0.2558	0.2062

Table 10: The lower bound of $P[R > k]$ calculated with $l = k+5$ for $t = 2$ and $a = 2$. Here $M_0 \sim Binom(10000, 0.99)$ p-values are from the null distribution and the others are from a non-null distribution with $a = 2$ and $\sigma = 1$. Values are given for a range of b, the number of slides for each genes. BIB refers to balanced incomplete block design and Ref refers to reference designs as introduced in Chapter 3.

b	design	lower bound of $P[R > k]$ for $t = 4$									
		k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9	k=10
12	BIB	0.0242	0.0056	0.0014	0.0003	8.5E-05	2.2E-05	5.6E-06	1.4E-06	3.7E-07	9.7E-08
	Loop1	0.0219	0.0049	0.0011	0.0003	6.5E-05	1.6E-05	3.9E-06	9.8E-07	2.4E-07	6.1E-08
	Loop2	0.0257	0.0061	0.0015	0.0004	9.9E-05	2.6E-05	6.8E-06	1.8E-06	4.7E-07	1.3E-07
24	BIB	0.2191	0.1171	0.0620	0.0324	0.0167	0.0085	0.0043	0.0021	0.0010	0.0005
	Loop1	0.0930	0.0358	0.0139	0.0054	0.0021	0.0008	0.0003	0.0001	4.3E-05	1.6E-05
	Loop2	0.3200	0.1982	0.1211	0.0727	0.0429	0.0250	0.0143	0.0081	0.0045	0.0025
36	BIB	0.9359	0.9007	0.8592	0.8115	0.7582	0.7004	0.6395	0.5770	0.5142	0.4528
	Loop1	0.5714	0.4387	0.3293	0.2416	0.1735	0.1221	0.0842	0.0570	0.0380	0.0249
	Loop2	0.9871	0.9789	0.9682	0.9543	0.9367	0.9152	0.8894	0.8591	0.8245	0.7858

Table 11: The lower bound of $P[R > k]$ calculated with $l = k + 5$ for $t = 4$. Here $M_0 \sim Binom(10000, 0.99)$ p-values are from the null distribution and the others are from a non-null distribution with $a_{t-1} = (1, 2, 1)$ (loop1) or $a_{t-1} = (2, 1, 1)$ (loop2). Values are given for a range of b, the number of slides for each genes. BIB refers to balanced incomplete block design and Loop refers to Loop designs as introduced in Chapter 3.

	lower bound of $P[S > j\|m_1]$ for $t = 2$										
b	j=0	j=1	j=2	j=3	j=4	j=5	j=6	j=7	j=8	j=9	j=10
	a=1										
4	0.0026	1.4E-05	7.8E-08	4.7E-10	2.9E-12	2.1E-14	1.0E-15	-1.1E-15	1.0E-15	2.6E-15	2.0E-15
6	0.0061	7.2E-05	9.5E-07	1.3E-08	1.8E-10	2.5E-12	3.6E-14	2.9E-15	3.4E-15	2.6E-15	2.2E-15
8	0.0136	0.0003	9.5E-06	2.7E-07	7.7E-09	2.2E-10	6.2E-12	1.8E-13	9.2E-15	2.3E-15	5.3E-15
10	0.0278	0.0014	7.2E-05	3.8E-06	2.0E-07	1.1E-08	5.4E-10	2.7E-11	1.4E-12	7.3E-14	1.5E-14
15	0.1175	0.0208	0.0037	0.0006	0.0001	1.8E-05	2.9E-06	4.5E-07	6.7E-08	9.8E-09	1.4E-09
20	0.3286	0.1358	0.0548	0.0212	0.0079	0.0028	0.0010	0.0003	0.0001	3.1E-05	9.3E-06
	a=2										
4	0.0088	0.0002	2.9E-06	5.8E-08	1.2E-09	2.4E-11	4.9E-13	1.2E-14	5.6E-16	5.7E-15	4.6E-15
6	0.0404	0.0030	0.0002	2.1E-05	1.8E-06	1.5E-07	1.3E-08	1.1E-09	8.7E-11	7.1E-12	5.8E-13
8	0.1569	0.0396	0.0104	0.0027	0.0007	0.0002	4.6E-05	1.1E-05	2.7E-06	6.5E-07	1.5E-07
10	0.4557	0.2653	0.1549	0.0889	0.0499	0.0273	0.0146	0.0076	0.0039	0.0019	0.0009
	j=0	j=1	j=10	j=20	j=30	j=40	j=50	j=60	j=70	j=80	
15	0.9991	0.9985	0.9740	0.7634	0.2836	0.0278	0.0005				
20	1	1	1	1	1	0.9999	0.9881	0.7728	0.1790	0.0025	

Table 12: The lower bound of $P[S > j|m_1]$ calculated with $l = k + 5$ for $t = 2$. Here $M_0 = 9900$ p-values are from the null distribution and $M_1 = 100$ p-values are from a non-null distribution with $a = 1$ or 2. Values are given for a range of b, the number of slides for each genes.

come from a normal (control) group and the rest of the genes come from groups where treatments have a common effect a. That is, the p-values of $M_0 = m_0$ tests have the uniform distribution and the remaining p-values have the alternative distribution. For each simulation, the p-values of the 10000 genes were sorted and the FDR method was applied with $q^* = 0.1$ to determine the value of k - the number of rejections based on the ordered p-values. For each specific (t, b, a), the proportion of the 10000 simulations rejecting k hypotheses was used as the estimated probability of $P[R = k]$. $P[R > k]$ was calculated from $1 - \sum_0^k P[R = i]$. A set of parallel simulations were run for fixed m_1, and the percentage of $S = j$ was used to estimate $P[S = j|m_1]$.

Tables 13 to 16 illustrate for $P[R = k]$ the simulation results (SR) and the estimated upper bounds (UB) using theorem 4.3.1 with $l = k + 5$ for the BIB design. UB with $l = k + 10$ are slightly smaller than the UB with $l = k + 5$. They are equal using four decimal places for most cases and a few exceptions are listed in Table 16. Note that since the SR themselves are also estimates and the observed values are not exactly equal to the true probabilities, sometimes the UB are slightly smaller than the SR as the tables shown. Most of the time, the UB are close to the SR for $a = 0.5$ *or* 1 and small (t,b) combinations even with $l = k + 5$. But the differences become larger with bigger t/b/a, especially with bigger a. For $a = 2$ and big b, the UB are too generous and are not very informative anymore.

Tables 17 to 22 give the lower bounds (LB) with $l = k + 5$ for $P[R > k]$. They are very close to the SR - the difference is less than 0.01 for almost all cases listed here. Therefore, it is reasonable to use the lower bound LB as an approximation of $P[R > k]$ in the experiment design. Again, sometimes the LB are slightly higher than the SR since SR are observed values instead of true values. These tables also give the LB estimated from a Poisson approximation which will be discussed in the next section.

Tables 23 to 25 list the LB with $l = j + 5$ for $P[S > j|m_1]$. Table 23 contains the LB and SR for different m_1, and the same m_1 but with different m. It shows that the LB are quite close to SR most of the time and the differences become bigger for large m_1, say 1000, and large m, say 100000. Moreover, as expected, for fixed m_1, $P[S > j|m_1]$ decreases for larger m; for fixed m, $P[S > j|m_1]$ increases with m_1; and

of course $P[S > j|m_1]$ increases with a. The LB have similar trends.

Tables 24 and 25 show that the LB are quite close to the SR and they can be used as good approximations of $P[S > j|m_1]$. They also indicate that for $a = 0.5$ *or* 1, the probability of having at least one true hypothesis is very small unless the number of slides, b, is large. The LB in these tables are sometimes larger than the LB for $P[R > k]$. The reason is that they are calculated from different settings: for $P[S > j|m_1]$, $m_1 = 100$ is fixed; while for $P[R > k]$, $M_1 \sim Binom(10000, 0.01)$ where M_1 might be smaller than 100.

The comparisons for the reference design contain similar messages and are not shown here.

4.5 Poisson Approximation

The cumbersome lower bound expression of $P[R > k]$ could be simplified with a Poisson approximation for moderate (t, b, a) as follows. The motivation is that the calculation in $P[R > k]$ involves counting measures on distinct intervals. For each fixed interval, the number of p-values falling into it has a binomial distribution, which of course could be approximated by a Poisson distribution for large m and a small success rate (probability that a p-value falls into this interval). By Corollary A.0.4 in Appendix A, the sequence of point processes N_n such that $N_n(a) = \sum_{i=1}^{\infty} I(p_i \leq \frac{a}{n})$ converges weakly to a Poisson process with mean measure $g\mu$, where μ denotes Lebesgue measure and $g = \lambda + (1 - \lambda)d_\delta(0)$. Here, $d_\delta(x)$ is the density function of $D(x)$. Hence, the number of $p_i \leq x/m$ for $i = 1, \cdots, m$ has an asymptotic Poisson distribution with mean $gx > 0$. Therefore, we have

$$
\begin{aligned}
& P[R \leq k] \\
\leq\ & P[N(0, \frac{k+1}{m}q^*) \leq k, \cdots, N(0, \frac{l}{m}q^*) \leq l-1] \\
=\ & \sum_{S2} P[N(0, \frac{k+1}{m}q^*) = r_1, N(\frac{k+1}{m}q^*, \frac{k+2}{m}q^*) = r_2, \cdots, N(\frac{l-1}{m}q^*, \frac{l}{m}q^*) = r_{l-k}] \\
=\ & \sum_{S2} P[N(0, \frac{k+1}{m}q^*) = r_1] \cdots P[N(\frac{l-1}{m}q^*, \frac{l}{m}q^*) = r_{l-k}]
\end{aligned}
$$

$$\approx \sum_{S2}[\frac{(g(k+1)q^*)^{r_1}}{r_1!}e^{-gq^*(k+1)} \cdot \frac{(gq^*)^{r_2}e^{-gq^*}}{r_2!} \cdots \frac{(gq^*)^{r_{l-k}}e^{-gq^*}}{r_{l-k}!}]$$

$$\approx \sum_{S2} \frac{(gq^*)^{\sum_i^{l-k} r_i}(k+1)^{r_1}}{r_1!r_2!\cdots r_{l-k}!}e^{-gq^*l}.$$

Here $S2$ was denoted in section 4.3.1. $d_\delta(0)$ could be approximated by a value of $d_\delta(x)$ at point x such that $F_0^{-1}(1-x)$ is very large as stated in the preceding chapter.

The numerical assessment in Tables 17 to 22 shows that for moderate values of (t, b, a) (which lead to a moderate non-centrality parameter δ), this approximated lower bound is close to the actual $P[R > k]$ with $l = k + 5$. The increase of the lower bound is very small from $l = k + 5$ to $l = k + 10$ (not shown). For large δ such that $g = \lambda + (1 - \lambda) * d_\delta(0) \geq 1.5$, this approximation is not good anymore since the error term $O(\frac{d_\delta(0)}{m})$ is large even for $m = 10000$. Thus, somewhat surprisingly for a small interval $(0, \frac{1}{m})$, the Poisson approximation is not good although m is big and the $m \cdot mean \leq 5$ $(m \cdot (\frac{1}{m} \cdot g) = 1.5)$. Hence, caution should be used when applying the Poisson approximation.

4.6 Example Data Set

We applied the FDR method on the HIV data set (illustrated in Table 1). The details will be discussed in Chapter 6. Applying the FDR method, we conclude that 14 genes are differentially expressed controlling FDR at level $q^* = 0.05$, and that 33 genes are differentially expressed controlling FDR at level $q^* = 0.1$. This is a relatively large number of rejections in a microarray experiment with 6 replicates according to either the simulated $P[R > k]$ or the lower bounds and these 14 genes are worthy of further study.

t	b		Upper bound of $P[R = k]$ with $a = 0.5$					
			k=1	k=2	k=3	k=4	k=5	k=6
2	4	Simu	0.0804	0.0123	0.0023	0.0002	0.0002	
		UB: l=k+5	0.0817	0.0297	0.0091	0.0026	0.0007	0.0002
2	6	Simu	0.0800	0.0158	0.0031	0.0009	0.0001	
		UB: l=k+5	0.0821	0.0300	0.0092	0.0027	0.0007	0.0002
2	8	Simu	0.0764	0.0155	0.0030	0.0005	0.0001	
		UB: l=k+5	0.0827	0.0305	0.0095	0.0028	0.0008	0.0002
3	6	Simu	0.0768	0.0152	0.0026	0.0012		
		UB: l=k+5	0.0816	0.0296	0.0091	0.0026	0.0007	0.0002
3	9	Simu	0.0789	0.0153	0.0027	0.0007	0.0001	
		UB: l=k+5	0.0819	0.0298	0.0092	0.0026	0.0007	0.0002
3	12	Simu	0.0841	0.0172	0.0027	0.0007		
		UB: l=k+5	0.0823	0.0301	0.0093	0.0027	0.0008	0.0002
4	12	Simu	0.0825	0.0156	0.0030	0.0008		
		UB: l=k+5	0.0818	0.0298	0.0091	0.0026	0.0007	0.0002
4	24	Simu	0.0873	0.0166	0.0033	0.0010	0.0001	0.0001
		UB: l=k+5	0.0832	0.0308	0.0096	0.0028	0.0008	0.0002
4	36	Simu	0.0873	0.0172	0.0027	0.0011	0.0004	0.0001
		UB: l=k+5	0.0858	0.0327	0.0105	0.0031	0.0009	0.0003
6	30	Simu	0.0834	0.0158	0.0022	0.0009	0.0002	
		UB: l=k+5	0.0823	0.0301	0.0093	0.0027	0.0007	0.0002
6	60	Simu	0.0844	0.0176	0.0050	0.0007		
		UB: l=k+5	0.0852	0.0322	0.0102	0.0030	0.0009	0.0002
6	90	Simu	0.0878	0.0212	0.0038	0.0011	0.0001	
		UB: l=k+5	0.0908	0.0362	0.0121	0.0038	0.0011	0.0003
8	56	Simu	0.0847	0.0153	0.0035	0.0005		
		UB: l=k+5	0.0828	0.0305	0.0095	0.0027	0.0008	0.0002
8	112	Simu	0.0837	0.0183	0.0041	0.0005	0.0001	
		UB: l=k+5	0.0872	0.0336	0.0109	0.0033	0.0010	0.0003
8	168	Simu	0.0937	0.0201	0.0044	0.0010	0.0002	
		UB: l=k+5	0.0965	0.0404	0.0141	0.0046	0.0014	0.0004
10	90	Simu	0.0812	0.0153	0.0033	0.0009	0.0001	
		UB: l=k+5	0.0833	0.0308	0.0096	0.0028	0.0008	0.0002
10	180	Simu	0.0876	0.0164	0.0036	0.0012	0.0003	
		UB: l=k+5	0.0894	0.0351	0.0116	0.0036	0.0010	0.0003
10	270	Simu	0.1001	0.0186	0.0049	0.0012	0.0004	0.0001
		UB: l=k+5	0.1029	0.0454	0.0165	0.0056	0.0018	0.0006

Table 13: Simulated values of $P[R = k]$ and its upper bound with $a = 0.5$. The bound is calculated from theorem 4.3.1 with $l = k + 5$ for the BIB design. Ten thousand simulations were performed with $m = 10000$, $\sigma = 1$, $q^* = 0.1$, and $\lambda = 0.99$ (only 1% genes are true differentially expressed).

t	b		Upper bound of $P[R=k]$ with $a=1$								
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9
2	4	Simu	0.0803	0.0140	0.0022	0.0009	0.0003	0.0001			
		UB: l=k+5	0.0825	0.0303	0.0094	0.0027	0.0008	0.0002	5.6E-05	1.5E-05	4.0E-06
2	6	Simu	0.0829	0.0157	0.0042	0.0010	0.0003				
		UB: l=k+5	0.0847	0.0320	0.0102	0.0030	0.0009	0.0002	6.9E-05	1.9E-05	5.1E-06
2	8	Simu	0.0866	0.0159	0.0049	0.0010	0.0003	0.0003			
		UB: l=k+5	0.0893	0.0358	0.0121	0.0038	0.0012	0.0003	0.0001	2.9E-05	8.3E-06
3	6	Simu	0.0813	0.0159	0.0024	0.0005	0.0002				
		UB: l=k+5	0.0822	0.0300	0.0093	0.0027	0.0007	0.0002	5.5E-05	1.5E-05	3.8E-06
3	9	Simu	0.0822	0.0135	0.0035	0.0006					
		UB: l=k+5	0.0838	0.0313	0.0099	0.0029	0.0008	0.0002	6.3E-05	1.7E-05	4.6E-06
3	12	Simu	0.0849	0.0170	0.0030	0.0007	0.0004	0.0001			
		UB: l=k+5	0.0874	0.0341	0.0112	0.0034	0.0010	0.0003	8.3E-05	2.3E-05	6.5E-06
4	12	Simu	0.0826	0.0171	0.0031	0.0003	0.0004				
		UB: l=k+5	0.0835	0.0311	0.0097	0.0029	0.0008	0.0002	6.1E-05	1.7E-05	4.4E-06
4	24	Simu	0.1005	0.0247	0.0053	0.0021	0.0005	0.0001			
		UB: l=k+5	0.1001	0.0442	0.0162	0.0055	0.0018	0.0006	0.0002	5.5E-05	1.7E-05
4	36	Simu	0.1508	0.0453	0.0160	0.0066	0.0018	0.0007	0.0002		
		UB: l=k+5	0.1484	0.0957	0.0492	0.0228	0.0100	0.0042	0.0017	0.0007	0.0003
6	30	Simu	0.0938	0.0190	0.0038	0.0005	0.0001	0.0001			
		UB: l=k+5	0.0892	0.0353	0.0117	0.0036	0.0011	0.0003	9.0E-05	2.5E-05	7.1E-06
6	60	Simu	0.1454	0.0430	0.0177	0.0063	0.0024	0.0005	0.0001		
		UB: l=k+5	0.1479	0.0931	0.0465	0.0210	0.0089	0.0036	0.0014	0.0005	0.0002
6	90	Simu	0.2088	0.1297	0.0858	0.0481	0.0287	0.0160	0.0085	0.0071	0.0036
		UB:l=k+5	0.2116	0.2655	0.2468	0.1980	0.1445	0.0985	0.0637	0.0395	0.0236
8	56	Simu	0.0957	0.0255	0.0065	0.0011	0.0001				
		UB:l=k+5	0.0986	0.0424	0.0152	0.0050	0.0016	0.0005	0.0001	4.5E-05	1.3E-05
8	112	Simu	0.1929	0.0903	0.0404	0.0186	0.0079	0.0032	0.0021	0.0005	0.0001
		UB:l=k+5	0.1987	0.1814	0.1257	0.0764	0.0428	0.0226	0.0114	0.0056	0.0026
8	168	Simu	0.1118	0.1157	0.1150	0.1057	0.0887	0.0754	0.0653	0.0502	0.0352
		UB:l=k+5	0.1205	0.2397	0.3429	0.4154	0.4514	0.4528	0.4263	0.3809	0.3255

Table 14: Similar to table 13: upper bound for $P[R=k]$ with $a=1$.

t	b		Upper bound of $P[R=k]$ with $a=2$									
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9	k=10
2	4	Simu	0.0875	0.0157	0.0038	0.0009	0.0001	0.0002				
		UB:l=k+5	0.0863	0.0334	0.0109	0.0033	0.0010	0.0003	8.1E-05	2.3E-05	6.3E-06	1.8E-06
2	6	Simu	0.1066	0.0249	0.0071	0.0022	0.0007	0.0001	0.0002			
		UB:l=k+5	0.1046	0.0504	0.0204	0.0077	0.0028	0.0010	0.0003	0.0001	4.0E-05	1.4E-05
2	8	Simu	0.1536	0.0574	0.0284	0.0105	0.0040	0.0022	0.0008	0.0004	0.0002	0.0002
		UB:l=k+5	0.1521	0.1166	0.0732	0.0422	0.0231	0.0122	0.0063	0.0032	0.0016	0.0008
3	6	Simu	0.0818	0.0172	0.0044	0.0009	0.0003	0.0001				
		UB:l=k+5	0.0852	0.0325	0.0104	0.0031	0.0009	0.0003	7.3E-05	2.0E-05	5.5E-06	1.5E-06
3	9	Simu	0.0990	0.0245	0.0061	0.0017	0.0005	0.0002				
		UB:l=k+5	0.1025	0.0478	0.0187	0.0068	0.0024	0.0008	0.0003	8.9E-05	2.9E-05	9.4E-06
3	12	Simu	0.1453	0.0590	0.0246	0.0083	0.0040	0.0014	0.0007	0.0005	0.0001	0.0001
		UB:l=k+5	0.1518	0.1114	0.0663	0.0359	0.0184	0.0091	0.0044	0.0020	0.0009	0.0004
4	12	Simu	0.1020	0.0211	0.0061	0.0016	0.0009	0.0001				
		UB:l=k+5	0.1019	0.0469	0.0181	0.0065	0.0022	0.0007	0.0002	8E-05	3E-05	8E-06
		UB:l=k+10	0.1019	0.0469	0.0181	0.0065	0.0022	0.0007	0.0002	7.9E-05	2.5E-05	8.0E-06
4	24	Simu	0.1154	0.1049	0.0935	0.0909	0.0791	0.0649	0.0589	0.048	0.0407	0.0334
		UB:l=k+5	0.1146	0.2122	0.2961	0.3600	0.4004	0.4173	0.4131	0.3920	0.3586	0.3179
		UB:l=k+10	0.1122	0.2088	0.2924	0.3562	0.3970	0.4143	0.4105	0.3898	0.3569	0.3166
6	30	Simu	0.2004	0.1100	0.0607	0.0360	0.0185	0.0106	0.0058	0.0027	0.0009	0.0004
		UB:l=k+5	0.1990	0.2192	0.1851	0.1377	0.0945	0.0613	0.0380	0.0227	0.0132	0.0074
		UB:l=k+10	0.1988	0.2190	0.1850	0.1376	0.0945	0.0612	0.0380	0.0227	0.0132	0.0074
8	56	Simu	0.0763	0.0825	0.0830	0.0853	0.0813	0.0842	0.0703	0.0685	0.0565	0.051
		UB:l=k+5	0.0765	0.1645	0.2592	0.3499	0.4272	0.4843	0.5177	0.5270	0.5145	0.4843
		UB:l=k+10	0.0746	0.1615	0.2555	0.3459	0.4232	0.4806	0.5143	0.5241	0.5121	0.4823

Table 15: Similar to table 13: upper bound for $P[R=k]$ with $a=2$. The upper bounds with $l=k+10$ are not shown for the first 6 blocks since they are same as the bounds for $l=k+5$ with four decimal places.

			Upper bound of $P[R=k]$ with $a=2$									
t	b		k=30	k=31	k=32	k=33	k=34	k=35	k=36	k=37	k=38	k=39
4	36	simu	0.0192	0.0206	0.0204	0.0229	0.0282	0.0302	0.0329	0.0321	0.0372	0.0385
		UB:l=k+5	0.5799	0.6724	0.7704	0.8722	0.9761	1	1	1	1	1
		UB:l=k+10	0.5745	0.6668	0.7644	0.8660	0.9697	1	1	1	1	1
6	60	simu	0.012	0.0119	0.0159	0.018	0.0184	0.0207	0.026	0.0292	0.0325	0.0306
		UB:l=k+5	0.3392	0.4116	0.4929	0.5828	0.6805	0.7848	0.8943	1	1	1
		UB:l=k+10	0.3366	0.4087	0.4897	0.5793	0.6767	0.7808	0.8901	1	1	1
t	b		k=90	k=91	k=92	k=93	k=94	k=95	k=96	k=97	k=98	k=99
6	90	simu	0.032	0.0333	0.0356	0.0356	0.0334	0.0346	0.0352	0.0341	0.0302	0.0299
		UB:l=k+5	1	1	1	1	1	1	1	1	1	1
		UB:l=k+10	1	1	1	1	1	1	1	1	1	1
t	b		k=80	k=81	k=82	k=83	k=84	k=85	k=86	k=87	k=88	k=89
8	112	simu	0.0346	0.0352	0.0334	0.0334	0.0344	0.0326	0.0365	0.0281	0.0277	0.027
		UB:l=k+5	1	1	1	1	1	1	1	1	1	1
		UB:l=k+10	1	1	1	1	1	1	1	1	1	1
8	168	simu	0.0094	0.0113	0.0147	0.0173	0.0175	0.019	0.0225	0.0218	0.0246	0.0273
		UB:l=k+5	0.9954	1	1	1	1	1	1	1	1	1
		UB:l=k+10	0.9953	1	1	1	1	1	1	1	1	1
t	b		k=25	k=26	k=27	k=28	k=29	k=30	k=31	k=32	k=33	k=34
10	90	simu	0.0372	0.0335	0.0275	0.0273	0.0207	0.0195	0.0172	0.0119	0.0095	0.0087
		UB:l=k+5	0.9043	0.8433	0.7753	0.7029	0.6288	0.5552	0.4841	0.4170	0.3549	0.2986
		UB:l=k+10	0.9010	0.8405	0.7729	0.7009	0.6272	0.5539	0.4830	0.4161	0.3542	0.2980

Table 16: Similar to table 13: upper bound for $P[R=k]$ with $a=2$. Due to limitation of space, extremely small value of $P[R=k]$ are not displayed.

t	b		lower bound of $P[R > k]$ with $a = 0.5$								
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9
2	4	simu	0.0150	0.0027	0.0004	0.0002	1.1E-16				
		LB:Multinomial	0.0187	0.0039	0.0008	0.0002	4.2E-05	9.7E-06	2.2E-06	5.2E-07	1.2E-07
		LB:Poisson	0.0187	0.0039	0.0008	0.0002	4.2E-05	9.7E-06	2.3E-06	5.3E-07	1.2E-07
2	6	simu	0.0199	0.0041	0.0010	1E-04	0				
		LB:Multinomial	0.0189	0.0039	0.0009	0.0002	4.4E-05	1.0E-05	2.4E-06	5.5E-07	1.3E-07
		LB:Poisson	0.0189	0.0039	0.0009	0.0002	4.4E-05	1.0E-05	2.4E-06	5.5E-07	1.3E-07
2	8	simu	0.0191	0.0036	0.0006	0.0001	1.1E-16				
		LB:Multinomial	0.0193	0.0040	0.0009	0.0002	4.6E-05	1.1E-05	2.5E-06	5.9E-07	1.4E-07
		LB:Poisson	0.0193	0.0041	0.0009	0.0002	4.7E-05	1.1E-05	2.6E-06	6.1E-07	1.4E-07
3	6	simu	0.0190	0.0038	0.0012	0					
		LB:Multinomial	0.0187	0.0039	0.0008	0.0002	4.2E-05	9.6E-06	2.2E-06	5.2E-07	1.2E-07
		LB:Poisson	0.0187	0.0039	0.0008	0.0002	4.2E-05	9.7E-06	2.2E-06	5.2E-07	1.2E-07
3	9	simu	0.0188	0.0035	0.0008	0.0001	0				
		LB:Multinomial	0.0188	0.0039	0.0008	0.0002	4.3E-05	9.9E-06	2.3E-06	5.4E-07	1.3E-07
		LB:Poisson	0.0188	0.0039	0.0009	0.0002	4.3E-05	1.0E-05	2.3E-06	5.4E-07	1.3E-07
3	12	simu	0.0206	0.0034	0.0007	0					
		LB:Multinomial	0.0190	0.0040	0.0009	0.0002	4.5E-05	1.0E-05	2.4E-06	5.6E-07	1.3E-07
		LB:Poisson	0.0191	0.0040	0.0009	0.0002	4.5E-05	1.1E-05	2.5E-06	5.8E-07	1.4E-07
4	12	simu	0.0194	0.0038	0.0008	0					
		LB:Multinomial	0.0188	0.0039	0.0008	0.0002	4.3E-05	9.8E-06	2.3E-06	5.3E-07	1.2E-07
		LB:Poisson	0.0188	0.0039	0.0008	0.0002	4.3E-05	9.9E-06	2.3E-06	5.4E-07	1.3E-07
4	24	simu	0.0211	0.0045	0.0012	0.0002	1.0E-04	1.1E-16			
		LB:Multinomial	0.0195	0.0041	0.0009	0.0002	4.7E-05	1.1E-05	2.6E-06	6.1E-07	1.5E-07
		LB:Poisson	0.0204	0.0044	0.0010	0.0002	5.5E-05	1.3E-05	3.2E-06	7.7E-07	1.9E-07
4	36	simu	0.0215	0.0043	0.0016	0.0005	1.0E-04	0			
		LB:Multinomial	0.0208	0.0045	0.0010	0.0002	5.6E-05	1.3E-05	3.2E-06	7.7E-07	1.9E-07
		LB:Poisson	0.0306	0.0080	0.0022	0.0006	0.0002	5.0E-05	1.5E-05	4.3E-06	1.3E-06

Table 17: Simulated result and lower bounds for $P[R > k]$ with $a = 0.5$. The bounds LB:multinomial are calculated from theorem 4.3.1 with $l = k + 5$ for the BIB design. The bounds LB:Poisson are calculated from a Poisson approximation as in section 4.5. Ten thousand simulations were performed with $m = 10000$, $\sigma = 1$, $q^* = 0.1$, and $\lambda = 0.99$ (only 1% genes are true differentially expressed).

t	b		lower bound of $P[R > k]$ with $a = 0.5$								
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9
6	30	simu	0.0191	0.0033	0.0011	0.0002	0				
		LB:Multinomial	0.0190	0.0040	0.0009	0.0002	4.4E-05	1.0E-05	2.4E-06	5.6E-07	1.3E-07
		LB:Poisson	0.0195	0.0041	0.0009	0.0002	4.8E-05	1.1E-05	2.6E-06	6.3E-07	1.5E-07
6	60	simu	0.0233	0.0057	0.0007	0					
		LB:Multinomial	0.0205	0.0044	0.0010	0.0002	5.3E-05	1.3E-05	3.0E-06	7.2E-07	1.7E-07
		LB:Poisson	0.0476	0.0152	0.0051	0.0017	0.0006	0.0002	7.59616E-05	2.7E-05	9.8E-06
6	90	simu	0.0262	0.0050	0.0012	0.0001	0				
		LB:Multinomial	0.0234	0.0053	0.0012	0.0003	7.2E-05	1.8E-05	4.4E-06	1.1E-06	2.8E-07
		LB:Poisson	0.9933	0.9948	0.9962	0.9972	0.9980	0.9986	0.9990	0.9993	0.9995
8	56	simu	0.0193	0.0040	0.0005	0					
		LB:Multinomial	0.0193	0.0040	0.0009	0.0002	4.6E-05	1.1E-05	2.5E-06	5.9E-07	1.4E-07
		LB:Poisson	0.0221	0.0050	0.0012	0.0003	6.9E-05	1.7E-05	4.3E-06	1.1E-06	2.8E-07
8	112	simu	0.0230	0.0047	0.0006	0.0001	1.1E-16				
		LB:Multinomial	0.0215	0.0047	0.0011	0.0003	5.9E-05	1.4E-05	3.5E-06	8.4E-07	2.1E-07
		LB:Poisson	0.9558	0.9570	0.9598	0.9631	0.9665	0.9697	0.9728 0.9756	0.9781	
8	168	simu	0.0257	0.0056	0.0012	0.0002	1.1E-16				
		LB:Multinomial	0.0264	0.0062	0.0015	0.0004	9.5E-05	2.4E-05	6.2E-06	1.6E-06	4.1E-07
		LB:Poisson	1	1	1	1	1	1	1	1	1
10	90	simu	0.0196	0.0043	0.0010	0.0001	1.1E-16				
		LB:Multinomial	0.0195	0.0041	0.0009	0.0002	4.7E-05	1.1E-05	2.6E-06	6.1E-07	1.5E-07
		LB:Poisson	0.0354	0.0099	0.0029	0.0009	0.0003	8.1E-05	2.5E-05	7.9E-06	2.5E-06
10	180	simu	0.0217	0.0053	0.0017	0.0005	0.0002	0			
		LB:Multinomial	0.0226	0.0050	0.0012	0.0003	6.7E-05	1.6E-05	4.0E-06	9.8E-07	2.4E-07
		LB:Poisson	1	1	1	1	1	1	1	1	1
10	270	simu	0.0252	0.0066	0.0017	0.0005	0.0001	1.1E-16			
		LB:Multinomial	0.0301	0.0074	0.0019	0.0005	0.0001	3.3E-05	8.8E-06	2.3E-06	6.2E-07
		LB:Poisson									

Table 18: Continuation of table 17. The last row of LB:Poisson is not given since f_0 in the denominator is very close to 0 and the lower bound cannot be calculated.

t	b		lower bound of $P[R > k]$ with $a = 1$								
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9
2	4	simu	0.0175	0.0035	0.0013	0.0004	1.0E-04	0			
		LB:Multinomial	0.0191	0.0040	0.0009	0.0002	4.5E-05	1.1E-05	2.5E-06	5.8E-07	1.4E-07
		LB:Poisson	0.0191	0.0040	0.0009	0.0002	4.5E-05	1.1E-05	2.5E-06	5.8E-07	1.4E-07
2	6	simu	0.0212	0.0055	0.0013	0.0003	0				
		LB:Multinomial	0.0204	0.0044	0.0010	0.0002	5.4E-05	1.3E-05	3.1E-06	7.6E-07	1.8E-07
		LB:Poisson	0.0205	0.0044	0.0010	0.0002	5.5E-05	1.3E-05	3.2E-06	7.8E-07	1.9E-07
2	8	simu	0.0224	0.0065	0.0016	0.0006	0.0003	0			
		LB:Multinomial	0.0232	0.0053	0.0013	0.0003	7.6E-05	1.9E-05	4.8E-06	1.2E-06	3.2E-07
		LB:Poisson	0.0239	0.0055	0.0013	0.0003	8.6E-05	2.2E-05	5.7E-06	1.5E-06	3.9E-07
3	6	simu	0.0190	0.0031	0.0007	0.0002	1.1E-16				
		LB:Multinomial	0.0190	0.0039	0.0009	0.0002	4.4E-05	1.0E-05	2.4E-06	5.6E-07	1.3E-07
		LB:Poisson	0.0190	0.0040	0.0009	0.0002	4.4E-05	1.0E-05	2.4E-06	5.6E-07	1.3E-07
3	9	simu	0.0176	0.0041	0.0006	0					
		LB:Multinomial	0.0199	0.0042	0.0009	0.0002	5.0E-05	1.2E-05	2.8E-06	6.8E-07	1.6E-07
		LB:Poisson	0.0200	0.0043	0.0010	0.0002	5.2E-05	1.2E-05	2.9E-06	7.1E-07	1.7E-07
3	12	simu	0.0212	0.0042	0.0012	0.0005	1E-04	0			
		LB:Multinomial	0.0219	0.0049	0.0011	0.0003	6.5E-05	1.6E-05	3.9E-06	9.7E-07	2.4E-07
		LB:Poisson	0.0230	0.0053	0.0013	0.0003	7.7E-05	2.0E-05	5.0E-06	1.3E-06	3.3E-07
4	12	simu	0.0209	0.0038	0.0007	0.0004	1.1E-16				
		LB:Multinomial	0.0197	0.0042	0.0009	0.0002	4.9E-05	1.1E-05	2.7E-06	6.5E-07	1.5E-07
		LB:Poisson	0.0199	0.0042	0.0009	0.0002	5.1E-05	1.2E-05	2.9E-06	6.9E-07	1.7E-07
4	24	simu	0.0327	0.0080	0.0027	0.0006	0.0001	1.1E-16			
		LB:Multinomial	0.0294	0.0073	0.0019	0.0005	0.0001	3.5E-05	9.4E-06	2.5E-06	6.8E-07
		LB:Poisson	0.0748	0.0294	0.0120	0.0050	0.0021	0.0009	0.0004	0.0002	7.7E-05
4	36	simu	0.0706	0.0253	0.0093	0.0027	0.0009	0.0002	0		
		LB:Multinomial	0.0730	0.0251	0.0088	0.0031	0.0011	0.0004	0.0001	4.4E-05	1.5E-05
		LB:Poisson	1	1	1	1	1	1	1	1	1

Table 19: Similar to table 17, with $a = 1$.

t	b		lower bound of $P[R > k]$ with $a = 1$								
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9
6	30	simu	0.0235	0.0045	0.0007	0.0002	1E-04	0			
		LB:Multinomial	0.0227	0.0051	0.0012	0.0003	6.9E-05	1.7E-05	4.2E-06	1.1E-06	2.6E-07
		LB:Poisson	0.0358	0.0100	0.0029	0.0009	0.0003	8.4E-05	2.6E-05	8.3E-06	2.6E-06
6	60	simu	0.0700	0.0270	0.0093	0.0030	0.0006	0.0001	1.1E-16		
		LB:Multinomial	0.0700	0.0234	0.0079	0.0027	0.0009	0.0003	0.0001	3.4E-05	1.1E-05
		LB:Poisson	1.0000	1.0000	1.0000	1.0000	1	1	1	1	1
6	90	simu	0.3307	0.2010	0.1152	0.0671	0.0384	0.0224	0.0139	0.0068	0.0032
		LB:Multinomial	0.3285	0.1961	0.1140	0.0646	0.0358	0.0194	0.0103	0.0054	0.0028
		LB:Poisson	1	1	1	1	1	1	1	1	1
8	56	simu	0.0332	0.0077	0.0012	0.0001	1.1E-16				
		LB:Multinomial	0.0279	0.0067	0.0017	0.0004	0.0001	2.9E-05	7.6E-06	2.0E-06	5.2E-07
		LB:Poisson	0.7006	0.6551	0.6209	0.5934	0.5703	0.5504	0.5328	0.5171	0.5029
8	112	simu	0.1632	0.0729	0.0325	0.0139	0.0060	0.0028	0.0007	0.0002	0.0001
		LB:Multinomial	0.1668	0.0761	0.0342	0.0152	0.0066	0.0028	0.0012	0.0005	0.0002
		LB:Poisson	1.0000	1.0000	1.0000	1.0000	1	1	1	1	1
8	168	simu	0.7396	0.6239	0.5089	0.4032	0.3145	0.2391	0.1738	0.1236	0.0884
		LB:Multinomial	0.7308	0.6120	0.4987	0.3957	0.3060	0.2310	0.1704	0.1230	0.0870
		LB:Poisson	1.0000	1.0000	1.0000	1.0000	1	1	1	1	1
10	90	simu	0.0320	0.0078	0.0015	0.0001	0				
		LB:Multinomial	0.0356	0.0093	0.0025	0.0007	0.0002	5.3E-05	1.5E-05	4.1E-06	1.1E-06
		LB:Poisson	1	1	1	1	1	1	1	1	1
10	180	simu	0.3302	0.1966	0.1138	0.0644	0.0359	0.0182	0.0102	0.0052	0.0023
		LB:Multinomial	0.3330	0.1979	0.1143	0.0643	0.0352	0.0189	0.0099	0.0051	0.0026
		LB:Poisson	1	1	1	1	1	1	1	1	1
10	270	simu	0.9509	0.9160	0.8759	0.8239	0.7632	0.6999	0.6287	0.5565	0.4889
		LB:Multinomial	0.9521	0.9189	0.8767	0.8259	0.7673	0.7025	0.6336	0.5626	0.4919
		LB:Poisson									

Table 20: Continuation of table 19. The last row of LB:Poisson is not given since f_0 in the denominator is very close to 0 and the lower bound cannot be calculated.

t	b		lower bound of $P[R > k]$ with $a = 2$								
			k=1	k=2	k=3	k=4	k=5	k=6	k=7	k=8	k=9
2	4	simu	0.0207	0.0050	0.0012	0.0003	0.0002	0			
		LB:Multinomial	0.0214	0.0047	0.0011	0.0003	6.3E-05	1.5E-05	3.8E-06	9.4E-07	2.3E-07
		LB:Poisson	0.0214	0.0047	0.0011	0.0003	6.3E-05	1.5E-05	3.8E-06	9.4E-07	2.4E-07
2	6	simu	0.0352	0.0103	0.0032	0.0010	0.0003	0.0002	0		
		LB:Multinomial	0.0347	0.0095	0.0027	0.0008	0.0002	7.0E-05	2.1E-05	6.4E-06	2.0E-06
		LB:Poisson	0.0360	0.0101	0.0030	0.0009	0.0003	8.6E-05	2.7E-05	8.5E-06	2.7E-06
2	8	simu	0.1042	0.0468	0.0184	0.0079	0.0039	0.0017	0.0009	0.0005	0.0003
		LB:Multinomial	0.1014	0.0431	0.0188	0.0082	0.0036	0.0016	0.0007	0.0003	0.0001
		LB:Poisson	0.1478	0.0779	0.0426	0.0238	0.0135	0.0077	0.0045	0.0026	0.0015
3	6	simu	0.0229	0.0057	0.0013	0.0004	0.0001	1.1E-16			
		LB:Multinomial	0.0207	0.0045	0.0010	0.0002	5.7E-05	1.4E-05	3.3E-06	8.1E-07	2.0E-07
		LB:Poisson	0.0208	0.0045	0.0010	0.0002	5.7E-05	1.4E-05	3.4E-06	8.2E-07	2.0E-07
3	9	simu	0.0330	0.0085	0.0024	0.0007	0.0002	1.1E-16			
		LB:Multinomial	0.0325	0.0086	0.0024	0.0007	0.0002	5.4E-05	1.6E-05	4.6E-06	1.3E-06
		LB:Poisson	0.0357	0.0100	0.0029	0.0009	0.0003	8.4E-05	2.6E-05	8.2E-06	2.6E-06
3	12	simu	0.0988	0.0398	0.0152	0.0069	0.0029	0.0015	0.0008	0.0003	0.0002
		LB:Multinomial	0.0929	0.0373	0.0152	0.0063	0.0026	0.0011	0.0004	0.0002	7.1E-05
		LB:Poisson	0.2042	0.1229	0.0765	0.0486	0.0313	0.0203	0.0133	0.0088	0.0058
4	12	simu	0.0298	0.0087	0.0026	0.0010	1E-04	0			
		LB:Multinomial	0.0317	0.0082	0.0022	0.0006	0.0002	4.9E-05	1.4E-05	3.9E-06	1.1E-06
		LB:Poisson	0.0377	0.0108	0.0033	0.0010	0.0003	1.0E-04	3.2E-05	1.0E-05	3.3E-06
4	24	simu	0.7153	0.6104	0.5169	0.4260	0.3469	0.282	0.2231	0.1751	0.1344
		LB:Multinomial	0.7086	0.6048	0.5083	0.4200	0.3414	0.2730	0.2148	0.1665	0.1271
		LB:Poisson	1	1	1	1	1	1	1	1	1
6	30	simu	0.2466	0.1366	0.0759	0.0399	0.0214	0.0108	0.005	0.0023	0.0014
		LB:Multinomial	0.2454	0.1361	0.0746	0.0402	0.0214	0.0112	0.0058	0.0029	0.0015
		LB:Poisson1	1	1	1	1	1	1	1	1	1
8	56	simu	0.8371	0.7546	0.6716	0.5863	0.505	0.4208	0.3505	0.282	0.2255
		LB:Multinomial	0.8322	0.7517	0.6670	0.5810	0.4969	0.4173	0.3443	0.2791	0.2225
		LB:Poisson	1	1	1	1	1	1	1	1	1
10	90	simu	0.9964	0.9926	0.9876	0.9816	0.9725	0.9601	0.9441	0.9267	0.9032
		LB:Multinomial	0.9960	0.9928	0.9882	0.9817	0.9727	0.9608	0.9454	0.9262	0.9028
		LB:Poisson	1	1	1	1	1	1	1	1	1

Table 21: Similar to table 20 with $a = 2$.

			lower bound of $P[R > k]$ with $a = 2$								
t	b		k=15	k=20	k=25	k=30	k=35	k=40	k=45	k=50	k=55
4	36	simu	0.9979	0.9882	0.9581	0.8851	0.7628	0.5826	0.3914	0.2265	0.1100
		LB:Multinomial	0.9974	0.9875	0.9559	0.8819	0.7511	0.5734	0.3840	0.2227	0.1112
		LB:Poisson	1	1	1	1	1	1	1	1	1
6	60	simu	0.9994	0.9965	0.9853	0.9442	0.8593	0.7098	0.5305	0.3362	0.1812
		LB:Multinomial	0.9994	0.9962	0.9829	0.9427	0.8544	0.7083	0.5220	0.3353	0.1858
		LB:Poisson	1	1	1	1	1	1			
t	b		k=60	k=70	k=80	k=90	k=100				
6	90	simu	0.9989	0.9812	0.8807	0.624	0.2915				
		LB:Multinomial	0.9989	0.9821	0.8809	0.6165	0.2892				
t	b		k=50	k=60	k=70	k=80	k=90	k=100			
8	112	simu	0.9967	0.9686	0.8217	0.5178	0.2047	0.0476			
		LB:Multinomial	0.9977	0.9686	0.8235	0.5140	0.2047	0.0490			
t	b		k=70	k=80	k=90	k=100	k=110	k=120			
8	168	simu	0.9994	0.9921	0.9317	0.7293	0.4108	0.1422			
		LB:Multinomial	0.9997	0.9924	0.9331	0.7274	0.3987	0.1380			
t	b		k=60	k=70	k=80	k=90					
10	180	simu	0.9937	0.9411	0.7515						
		LB:Multinomial	0.9940	0.9422	0.7499						
t	b		k=80	k=90	k=100	k=110					
10	270	simu	0.9956	0.9589	0.8066	0.4920					
		LB:Multinomial	0.9966	0.9610	0.8037	0.4924					

Table 22: Continuation of table 21. Representative k values are chosen to show the distribution. The last few rows of LB:Poisson are not given since g is large and g^x with $x > 50$ cannot be calculated.

m	m_1	t	b	a		lower bound of $P[S > j\|m_1]$ j=0	j=1	j=2	j=3	j=4
1000	50	4	12	0.5	simu	0.0098	1E-04	0		
					LB: l=j+5	0.0078	0.0001	1.9E-06	3.1E-08	5.2E-10
1000	100	4	12	0.5	simu	0.017	0.0008	0		
					LB:l=j+5	0.0155	0.0005	1.5E-05	5.2E-07	1.8E-08
10000	100	4	12	0.5	simu	0.0016	0			
					LB:l=j+5	0.0016	0.0000	1.7E-08	6.1E-11	2.2E-13
10000	500	4	12	0.5	simu	0.0094	0.0002	0		
					LB:l=j+5	0.0079	0.0001	2.1E-06	3.9E-08	7.3E-10
100000	100	4	12	0.5	simu	0.0002	0			
					LB:l=j+5	5.1E-08	1.8E-11	6.4E-15	1.1E-16	
10000	1000	4	12	0.5	simu	0.0192	0.0009	0		
					LB:l=j+5	0.0005	1.7E-05	6.2E-07	2.3E-08	
100000	1000	4	12	0.5	simu	0.0017				
					LB:l=j+5		5.1E-06	1.8E-08	6.9E-11	2.7E-13
1000	50	4	12	1	simu	0.0258	0.0010	0		
					LB:l=j+5	0.0197	0.0007	2.8E-05	1.1E-06	4.2E-08
1000	100	4	12	1	simu	0.0468	0.0032	0.0002	0	
					LB:l=j+5	0.0394	0.0028	0.0002	1.7E-05	1.4E-06
10000	100	4	12	1	simu	0.0049	0.0002	1E-04		
					LB:l=j+5	0.0042	3.4E-05	3.0E-07	2.7E-09	2.5E-11
10000	500	4	12	1	simu	0.0236	0.0013	1E-04	0	
					LB:l=j+5	0.0211	0.0008	3.7E-05	1.7E-06	8.0E-08
100000	100	4	12	1	simu	0.0006	0			
					LB: l=j+5	3.7E-07	3.5E-10	3.4E-13	6.7E-16	
10000	1000	4	12	1	simu	0.0491	0.0042	1E-04	0	
					LB:l=j+5	0.0033	0.0003	2.6E-05	2.4E-06	
100000	1000	4	12	1	simu	0.0048	0.0002	0		
					LB:l=j+5		3.8E-05	3.6E-07	3.6E-09	3.7E-11
1000	50	4	12	2	simu	0.1674	0.0409	0.0104	0.003	0.0006
					LB:l=j+5	0.1433	0.0319	0.0073	0.0016	0.0004
1000	100	4	12	2	simu	0.3113	0.1371	0.0593	0.0257	0.0121
					LB:l=j+5	0.2794	0.1137	0.0474	0.0197	0.0080
10000	100	4	12	2	simu	0.0426	0.0047	0.0003	0	
					LB:l=j+5	0.0350	0.0022	0.0001	9.5E-06	6.3E-07
10000	500	4	12	2	simu	0.1996	0.0655	0.0245	0.0076	0.0027
					LB:l=j+5	0.1726	0.0487	0.0146	0.0044	0.0014
100000	100	4	12	2	simu	0.0059	1E-04	0		
					LB:l=j+5	2.9E-05	2.3E-07	1.8E-09	1.5E-11	
10000	1000	4	12	2	simu	0.3819	0.2054	0.115	0.067	0.0405
					LB:l=j+5	0.1681	0.0879	0.0466	0.0249	
100000	1000	4	12	2	simu	0.0437	0.0041	1E-04	0	
					LB:l=j+5		0.0028	0.0002	1.8E-05	1.5E-06

Table 23: Simulated values of $P[S > j|m_1]$ and its lower bound, for different m and different m_1. For $m = 100000$, SAS cannot compute the lower bound of $P[S > 0|m_1]$ since it cannot compute $finv(0.1/100000)$ which is involved in the calculation.

t	b	a		lower bound of $P[S > j\|m_1]$					
				j=0	j=1	j=2	j=3	j=4	j=5
2	4	1	simu	0.0028	0.0001	0			
			LB:l=j+5	0.0026	0.0000	7.8E-08	4.7E-10	2.9E-12	2.1E-14
2	4	2	simu	0.0110	0.0002	0			
			LB:l=j+5	0.0088	0.0002	2.9E-06	5.8E-08	1.2E-09	2.4E-11
2	6	1	simu	0.0079	0.0001	0			
			LB:l=j+5	0.0061	0.0001	9.5E-07	1.3E-08	1.8E-10	2.5E-12
2	6	2	simu	0.0467	0.0044	0.0005	0		
			LB:l=j+5	0.0404	0.0030	2.5E-04	2.1E-05	1.8E-06	1.5E-07
2	8	1	simu	0.0158	0.0004	1.1E-16			
			LB:l=j+5	0.0136	0.0003	9.5E-06	2.7E-07	7.7E-09	2.2E-10
2	8	2	simu	0.1874	0.0526	0.0179	0.0054	0.0012	0.0005
			LB:l=j+5	0.1569	0.0396	0.0104	0.0027	0.0007	0.0002
4	24	0.5	simu	0.0048	0				
			LB:l=j+5	0.0037	2.6E-05	1.9E-07	1.4E-09	1.1E-11	8.3E-14
4	24	1	simu	0.0403	0.0022	0.0002	1E-04	0	
			LB:l=j+5	0.0311	0.0016	8.1E-05	4.2E-06	2.1E-07	1.0E-08
4	24	2	simu	0.8199	0.6938	0.5769	0.4654	0.3668	0.2783
			LB:l=j+5	0.7851	0.6462	0.5193	0.4053	0.3069	0.2253
4	36	0.5	simu	0.0098	0.0004	1.1E-16			
			LB:l=j+5	0.0078	0.0001	1.5E-06	2.1E-08	2.9E-10	4.0E-12
4	36	1	simu	0.1441	0.0296	0.0054	0.0009	0.0003	1E-04
			LB:l=j+5	0.1268	0.0220	0.0037	0.0006	9.4E-05	1.4E-05
6	30	1	simu	0.0152	0.0005	0			
			LB:l=j+5	0.0132	0.0003	6.9E-06	1.6E-07	3.7E-09	8.5E-11
6	30	2	simu	0.4030	0.1940	0.0901	0.0398	0.0169	0.0064
			LB:l=j+5	0.3629	0.1598	0.0680	0.0277	0.0108	0.0040
6	60	1	simu	0.1336	0.0230	0.0042	0.0006	0	
			LB:l=j+5	0.1235	0.0203	0.0032	0.0005	7.0E-05	9.6E-06
6	90	1	simu	0.5104	0.2798	0.1439	0.0712	0.0344	0.0145
			LB:l=j+5	0.4704	0.2407	0.1159	0.0524	0.0223	0.0090
8	56	0.5	simu	0.0032	0.0001	0			
			LB:l=j+5	0.0031	0.0000	1.1E-07	7.0E-10	4.4E-12	2.9E-14
8	56	1	simu	0.0337	0.0017	0.0002	0		
			LB:l=j+5	0.0282	0.0012	5.4E-05	2.3E-06	9.8E-08	4.0E-09
8	56	2	simu	0.9138	0.8336	0.7405	0.6404	0.5415	0.4445
			LB:l=j+5	0.8891	0.7948	0.6921	0.5850	0.4794	0.3806
8	112	0.5	simu	0.0112	0.0003	0			
			LB:l=j+5	0.0099	0.0002	2.7E-06	4.5E-08	7.4E-10	1.2E-11
8	112	1	simu	0.3070	0.1068	0.0348	0.0108	0.0025	0.0007
			LB:l=j+5	0.2809	0.0920	0.0284	0.0082	0.0022	0.0006
8	168	0.5	simu	0.0272	0.0011	0			
			LB:l=j+5	0.0246	0.0009	3.3E-05	1.2E-06	4.2E-08	1.4E-09
8	168	1	simu	0.8496	0.7123	0.5816	0.4542	0.3336	0.2371
			LB:l=j+5	0.8230	0.6763	0.5317	0.3990	0.2860	0.1960

Table 24: Simulated values of $P[S > j|m_1]$ and its lower bound, with $m = 10000$ and $m_1 = 100$.

			lower bound of $P[S > j \mid m_1]$ with $a = 2$								
t	b		j=15	j=20	j=25	j=30	j=35	j=40	j=45	j=50	j=55
4	36	simu	0.9994	0.9943	0.9637	0.8611	0.6520	0.3700	0.1422	0.0348	0.0059
		LB:l=j+5	0.9980	0.9868	0.9384	0.8019	0.5569	0.2826	0.0956	0.0199	0.0024
6	60	simu		0.9992	0.9907	0.9456	0.8100	0.5645	0.2810	0.0893	0.0174
		LB:l=j+5	0.9997	0.9973	0.9815	0.9156	0.7437	0.4684	0.2042	0.0560	0.0089
t	b		j=50	j=60	j=70	j=80	j=90				
6	90	simu		1	0.9995	0.8562	0.0583				
		LB:l=j+5		1	0.9987	0.8043	0.0413				
8	112	simu	0.9999	0.9888	0.6847	0.0632	0.0001				
		LB:l=j+5	0.9999	0.9812	0.6084	0.0407	1.8E-05				
8	168	simu					0.998				
		LB:l=j+5			1	1	0.9978				

Table 25: Similar to table 24 for $a = 2$ and large (t, b). Representative j values are chosen to show the distribution. The last few rows only show for j up to 90 since the calculation involves computing $comb(m_1, x)$ and SAS is only able to calculate for $x < 100$.

Chapter 5

Application of Westfall-Young Method under The Assumption of Independence

Multiplicity-adjusted p-values $\tilde{p}$ are sometimes computed to adjust for the high dimension problem in multiple tests. Westfall-Young [31] give the mathematical definition of an adjusted p-values, $\tilde{p}_i$, as $\tilde{p}_i = inf\{\alpha \mid H_i \; is \; rejected \; at \; FWER = \alpha\}$. Here FWER is the family-wise error rate - the probability of at least one Type I error in the family. The decision to reject a test is then made by comparing $\tilde{p}$ directly to a predefined level α, to reach a family-wise error rate FWER=α. The reasons to use adjusted p-values are that the decision is very easily made once the adjusted p-values have been calculated, and the usual cumbersome table look-ups can be avoided. The adjusting methods generally belong to either single step methods and step down methods. Single step methods perform equivalent multiplicity adjustments for all tests, regardless of the ordering of the observed p-values [31]. Both the Bonferroni and Sidak methods are single-step methods: Bonferroni's method, which rejects hypothesis H_i if $p_i \leq \alpha/m$, leads to the adjusted p-values $\tilde{p}_i = min(mp_i, 1)$; Sidak's method rejects H_i if $p_i \leq 1 - (1-\alpha)^{\frac{1}{m}}$, and its adjusted p-values are $\tilde{p}_i = 1 - (1-p_i)^m$. In contrast, step-down methods consider the order of the p-values in the adjustment. Holm's sequentially rejective algorithm is the step-down adjustment based on the

Bonferroni method: reject hypotheses $H_{(1)}, \cdots, H_{(i)}$ if i is the biggest number satisfying $p_{(i)} \leq \alpha/(m-i+1)$, where $p_{(1)}, \cdots, p_{(m)}$ are the ordered p-values. This algorithm leads to the adjusted p-values $\tilde{p}_{(i)} = max(\tilde{p}_{(i-1)}, (m-i+1)p_{(i)})$. Holm's method is still conservative since it is based on the Bonferroni inequality. A slightly less conservative adjustment is possible by using the Sidak inequality, which leads to $\tilde{p}_{(i)} = max(\tilde{p}_{(i-1)}, 1-(1-p_{(i)})^{(m-i+1)})$. All these methods control FWER at level α. The FDR method we discussed in the preceding chapter can also be viewed as a step-down method which adjusts p-values similar to Holm's method - it compares $p_{(i)}$ to i/mq^*. As we stated previously, it controls the FDR instead, which is smaller than or equal to the FWER. Thus, the FDR method is less conservative than both Holm's method and the step-down Sidak adjustment.

In this chapter, we will consider another step-down method - the Westfall and Young resampling procedure. This procedure has aroused interest recently in microarray data analysis. It controls FWER and is also less conservative than both Holm's method and the step-down Sidak adjustment since it considers the dependence structure among p-values using resampling. Unfortunately, it is impossible to involve an unknown dependence structure among p-values in experimental design and we will assume independence here. The Westfall-Young method is the same as the step-down Sidak method under the independence assumption. The p-values can be easily adjusted for the step-down Sidak method given the ordered observed p-values and thus the decision to reject can be made in a straightforward manner. However, during experimental design, there are no observed p-values. Simulations will be necessary to obtain the relationship between sample size (the number of slides) and the number of rejections. Thus, full sets of m p-values need to be simulated to calculate ordered p-values and a large number of full sets are needed to estimate the relation, which could be time consuming for large m. In section 1, we review the Westfall-Young method and propose a way to simulate the first few ordered p-values approximately but quickly. In section 2, we estimate the asymptotic distribution for the first few adjusted p-values. In section 3, we will discuss a method suggested by an anonymous reader which modifies the Westfall-Young adjustment.

5.1 Westfall-Young Method Under The Independence Assumption

We will briefly describe the original Westfall-Young method. Westfall-Young [31, page 66] proposed a (free) step-down adjustment to p-values by incorporating the dependence characteristics of p-values. Denote the p-values of m simultaneous tests by $p_1, p_2, \cdots, p_m$. Let the ordered p-values have indices $r_1, r_2, \cdots, r_m$, so that $p_{(1)} = p_{r_1}, p_{(2)} = p_{r_2}, \cdots, p_{(m)} = p_{r_m}$. Denote the random variable for the unadjusted p-value of the ith test H_i by $\mathbb{P}_i$. Define $H_0^C = \cap_{i=1}^m H_i^0 = \{H_0 \text{ are true for all genes}\}$. The Westfall-Young minP adjusted p-values can be calculated as follows:

$$
\begin{aligned}
\tilde{p}_{(1)} &= Pr(\min_{l \in \{r_1, r_2, \cdots, r_m\}} \mathbb{P}_l \leq p_{(1)} | H_0^C) \\
\tilde{p}_{(2)} &= \max[\tilde{p}_{(1)}, Pr(\min_{l \in \{r_2, \cdots, r_m\}} \mathbb{P}_l \leq p_{(2)} | H_0^C)] \\
&\vdots \\
\tilde{p}_{(k)} &= \max[\tilde{p}_{(k-1)}, Pr(\min_{l \in \{r_k, \cdots, r_m\}} \mathbb{P}_l \leq p_{(k)} | H_0^C)] \\
&\vdots \\
\tilde{p}_{(m)} &= \max[\tilde{p}_{(m-1)}, Pr(\mathbb{P}_{r_m} \leq p_{(m)} | H_0^C)]
\end{aligned}
$$

The adjusted p-values are monotonic and will be compared with a common significance level α to make the decision as to which to highlight. That is, let $\tilde{p}_{(k)}$ be the first i such that $\tilde{p}_{(i)} > \alpha$, then reject all of the corresponding null hypotheses $H_{(1)}^0, \cdots, H_{(k-1)}^0$.

The Westfall-Young method improves the Sidak step-down method in the way that it considers the dependence among p-values. However, it is impractical to have a specific dependence structure assumption in experimental design in general. Therefore, as in the preceding chapter, we assume that all p-values are independent of each other. Under this assumption, the Westfall-Young adjusted p-values are just the Sidak step-down adjusted p-values. The details are as follows. We will first quote a simulation algorithm from Westfall-Young [31, page 66] to illustrate the application

of their adjustment:

"**Algorithm 2.8 Free step-down resampling method**

0. Initialize counting variables: $COUNT_i = 0$, $i = 1, \cdots, m$.

1. Generate a vector $(p^*_{r_1}, \cdots, p^*_{r_m})$ from the same (or at least, approximately the same) distribution as the original p-values $(P_{r_1}, \cdots, P_{r_m})$ under the complete null hypotheses. Note that the sequence $\{r_j\}$ is fixed throughout the simulations. Thus the $p^*_{r_j}$ will not have the same monotonicity as the original p-values $p_{(j)}$.

2. Define the successive minima:

$$\begin{aligned} q^*_m &= p^*_{r_m} \\ q^*_{m-1} &= min(q^*_m, p^*_{r_{m-1}}) \\ q^*_{m-2} &= min(q^*_{m-1}, p^*_{r_{m-2}}) \\ &\vdots \\ q^*_1 &= min(q^*_2, p^*_{r_1}). \end{aligned}$$

3. If $q^*_i \leq p_{(i)}$, then $COUNT_i \leftarrow COUNT_i + 1$.

4. Repeat 1-3 N times. Compute $\tilde{p}^N_{(i)} = COUNT_i/N$.

5. Enforce monotonicity using successive maximization:

$$\begin{aligned} \tilde{p}^N_{(1)} &= \tilde{p}^N_{(1)} \\ \tilde{p}^N_{(2)} &\leftarrow max(\tilde{p}^N_{(1)}, \tilde{p}^N_{(2)}) \\ &\vdots \\ \tilde{p}^N_{(m)} &\leftarrow max(\tilde{p}^N_{(m-1)}, \tilde{p}^N_{(m)}) \end{aligned}$$

Once monotonicity has been enforced, the simulation-based estimates $\tilde{p}^N_{(j)}$ are reasonable approximations of the actual values $\tilde{p}_{(j)}$, provided N is sufficiently large."

According to this Algorithm, to calculate $\tilde{p}_{(k)}$, firstly the random variables $\mathbb{P}_{r_1}$, $\cdots$, $\mathbb{P}_{r_{k-1}}$ which have the original observed values $p_{r_1} = p_{(1)}, \cdots, p_{r_{k-1}} = p_{(k-1)}$ are

discarded. Secondly, the probability that the minimum of the random variables $\mathbb{P}_{r_j}$, $j \geq k$ is smaller than $p_{(k)}$ will be calculated ($P[min_{l\in\{r_k,\cdots,r_m\}}\mathbb{P}_l \leq p_{(k)} \mid H_0^C]$). That is, the cdf of the first order statistic of p-values among $r_k, \cdots, r_m$ at point $p_{(k)}$ under H_0^C. Finally, the maximum of this probability and the adjusted $\tilde{p}_{(k-1)}$ will be used as the adjusted $\tilde{p}_{(k)}$. Under H_0^C and the independence assumption, all p-values have i.i.d uniform distributions on [0,1]. Since the distribution function of the first order statistic of n i.i.d uniform random variable is $F_{(1)}(x) = \int_0^x n(1-y)^{n-1}dy = 1-(1-x)^n$, we have for the k^{th} adjusted p-values

$$\tilde{p}_{(k)} = \max[\tilde{p}_{(k-1)}, Pr(\min_{l\in\{r_k,\cdots,r_m\}}\mathbb{P}_l \leq p_{(k)}|H_0^C)],$$

the second term is actually

$$\begin{aligned}
& Pr(\min_{l\in\{r_k,\cdots,r_m\}}\mathbb{P}_l \leq p_{(k)}|H_0^C) \\
= \; & Pr(\text{the first order statistic of p-values among the (m-k+1) tests } \leq p_{(k)}|H_0^C) \\
= \; & \int_0^{p_{(k)}}(m-k+1)(1-x)^{m-k}dx \\
= \; & P[Beta(1, m-k+1) \leq p_{(k)}] \\
= \; & 1-(1-p_{(k)})^{m-k+1}.
\end{aligned}$$

These are the step-down Sidak adjusted p-values. The first k hypotheses would be rejected if $\tilde{p}_{(k+1)}$ is the first i such that $\tilde{p}_{(i)} > \alpha$ for a predefined FWER level α.

The step-down Sidak adjusted p-values are calculated based on ordered observed p-values. As we mentioned before, to study the relation between the number of rejection and the number of slides to be used for each gene in an experiment, simulations could be used to obtain the ordered "observed" p-values. That is, one needs to simulate all m p-values a large number of times, calculate $p_{(k)}$, then make the adjustments. These steps are time consuming and could be speeded up under the independence assumption. The key feature here is that the $mp_{(i)}$'s converge to a standard Poisson process. For $M_0 \sim Binom(m, \lambda)$ genes from H_0 and $m - M_0$ genes from H_1, we know the M_0 p-values have i.i.d U(0,1), and M_1 p-values have i.i.d alternative distributions $D_\delta(p)$ (defined in Chapter 3). Under the independence assumption, the intervals between successive points $m(p_{(i)} - p_{(i-1)})$ are asymptotically independent exponential

variables of mean $1/g$ as $m \to \infty$ with $g = \lambda + (1-\lambda) \cdot d_\delta(0)$ (A.0.4). Thus, for small k (relative to m) and independent $E_i \sim Exp(1/g)$, we have

$$mp_{(1)} \to E_1, \ mp_{(2)} \to E_1 + E_2, \ \cdots \ mp_{(k)} \to E_1 + E_2 + \cdots + E_k.$$

We could simulate only k independent E_i from $Exp(1/g)$ a large number of times, and calculate $\tilde{p}_{(k)} \approx max[\tilde{p}_{(k-1)}, 1 - (1 - \sum_1^k E_i/m)^{m-k+1}]$.

One could start with a reasonable predefined value of k. If all $\tilde{p}_{(k)} \leq \alpha$, one could simulate more $E_i \sim Exp(1/g)$ and combine with the previous simulated data to calculate $\tilde{p}_{(l)}$ for $l > k$ until $\tilde{p}_{(l)} > \alpha$. This way, the simulation will be accelerated and the time saved becomes significant when m is large.

A word of caution, as mentioned in the preceding chapter: the Poisson approximation with mean g of the mixed distribution is only good for moderate values of the non-centrality parameter δ, where the range of moderate values of δ is relative to the value of t and b. In general, it is safe to use the Poisson approximation for $g < 1.5$.

5.2 Distribution of The First k Adjusted P-values

We will look at the marginal distributions of the step-down Sidak adjusted p-values for small k relative to m. Under H_0^C: for $x \in [0, 1]$,

$$\begin{aligned} P[\tilde{p}_{(1)} \leq x] &= P[1 - (1 - p_{(1)})^m \leq x] \\ &= P[p_{(1)} \leq 1 - (1-x)^{\frac{1}{m}}] \\ &= \int_0^{1-(1-x)^{\frac{1}{m}}} m(1-t)^{t-1} dt \\ &= x. \end{aligned}$$

As expected, $\tilde{p}_{(1)}$ has a uniform distribution on $[0, 1]$. Of more interest is the distribution of $\tilde{p}_{(k)}$, and we will use the beta expression for simplicity:

$$P[\tilde{p}_{(k)} \leq x] = P[max(P(Beta(1, m) \leq p_{(1)}), \cdots, P(Beta(1, m-k+1) \leq p_{(k)})) \leq x]$$

Define $q(x)_i$ to be the x quantile of beta(1, m-i+1). That is, $F_{Beta(1,m-i+1)}(q(x)_i) = x$. Thus, for $1 \leq i \leq k$, $P(Beta(1, m-i+1) \leq p_{(i)}) = F_{Beta(1,m-i+1)}(p_{(i)}) \leq x$, which is equivalent to $p_{(i)} \leq q(x)_i$. Therefore, we have

$$\begin{aligned} P[\tilde{p}_{(k)} \leq x] &= P[P(Beta(1,m) \leq p_{(1)}) \leq x, \cdots, P(Beta(1,m-k+1) \leq p_{(k)}) \leq x] \\ &= P[p_{(1)} \leq q(x)_1, \cdots, p_{(k)} \leq q(x)_k] \end{aligned}$$

Obviously, the distribution of $\tilde{p}_{(k)}$ depends on the joint distribution of the observed ordered p-values $p_{(1)}, \cdots, p_{(k)}$. Even though under H_0^C, all the p_is have a marginal uniform distribution on [0,1], the joint distribution of the first k order statistics still has a cumbersome expression. It will be much easier to use an approximation. We know that the intervals between successive points are asymptotically independent exponential variables of mean 1, i.e. $mp_{(1)} \to E_1, m(p_{(2)} - p_{(1)}) \to E_2$, $m(p_{(k)} - p_{(k-1)}) \to E_k$, as $m \to \infty$, for $E_1, \cdots, E_k$ independent exponential random variables with mean 1. Therefore,

$$\begin{aligned} P[\tilde{p}_{(k)} \leq x] =& P[p_{(1)} \leq q(x)_1, \cdots, p_{(k)} \leq q(x)_k] \\ \approx& P[E_1 \leq mq(x)_1, \cdots, \sum_{i=1}^{k} E_i \leq mq(x)_k] \\ =& \int_0^{mq(x)_1} \cdots \int_0^{mq(x)_k - x_1 - \cdots - x_{k-1}} e^{-(x_1 + \cdots + x_k)} dx_1 \cdots dx_k. \end{aligned}$$

We will evaluate the integral for $k \leq 4$ for illustration:

$$\begin{aligned} P[\tilde{p}_{(1)} \leq x] \approx& 1 - e^{-mq(x)_1} \\ P[\tilde{p}_{(2)} \leq x] \approx& 1 - e^{-mq(x)_1} - mq(x)_1 e^{-mq(x)_2} \\ P[\tilde{p}_{(3)} \leq x] \approx& 1 - e^{-mq(x)_1} - mq(x)_1 e^{-mq(x)_2} - m^2 q(x)_1 e^{-mq(x)_3}(q(x)_2 - q(x)_1/2) \\ P[\tilde{p}_{(4)} \leq x] \approx& 1 - e^{-mq(x)_1} - mq(x)_1 e^{-mq(x)_2} - m^2 q(x)_1 e^{-mq(x)_3}(q(x)_2 - q(x)_1/2) \\ & - m^3 q(x)_1 e^{-mq(x)_4}(q(x)_2 q(x)_3 - q(x)_1 q(x)_3/2 + q^2(x)_1/6 - q^2(x)_2/2). \end{aligned} \tag{5.2.1}$$

In general, we have the following recursion formula:

$$P[\tilde{p}_{(k)} \leq x] \approx P[\tilde{p}_{(k-1)} \leq x] - e^{-mq(x)_k} \int_0^{mq(x)_1} \cdots \int_0^{mq(x)_k - x_1 - \cdots - x_{k-2}} 1 dx_1 \cdots dx_{k-1}.$$

	k=1	k=2	k=3	k=4
x=0.01	0.01	0.00005	1.7E-7	4.2E-10
x=0.05	0.05	0.0013	0.00002	2.8E-7
x=0.1	0.10	0.0052	0.0002	4.7E-6
x=0.2	0.20	0.0214	0.0016	0.0009

Table 26: Estimation of $P[\tilde{p}_{(k)} \leq x]$ using equation (5.2.1). It is assumed that 10000 genes were tested simultaneously and the null hypothesis is true for all of them.

Obviously, the distribution of $\tilde{p}_{(k)}$ for large k will have a messy expression although the integrals are not difficult to evaluate analytically. Simulations could be performed for large k to obtain the distribution. Table 26 illustrates the approximate value of $P[\tilde{p}_{(k)} \leq x]$ for $m = 10000$ and $k \leq 4$ using equation (5.2.1). Of course, we anticipate a uniform distribution for all unadjusted p-values. The table shows that the probability $P[\tilde{p}_{(k)} \leq x]$ is small for even k=3. Figure 9 plots the estimated distribution function $P[\tilde{p}_{(k)} \leq x]$ for $m = 10000$, $k = 1, 2, 3, 4$.

We can also compute an asymptotic joint distribution, e.g, for $k = 3$. Consider $(\tilde{p}_{(1)}, \tilde{p}_{(2)}, \tilde{p}_{(3)})$, for $x_1 \leq x_2 \leq x_3$, we have:

$$
\begin{aligned}
& P[\tilde{p}_{(1)} \leq x_1, \tilde{p}_{(2)} \leq x_2, \tilde{p}_{(3)} \leq x_3] \\
=& P[p_{(1)} \leq q(x_1)_1, p_{(2)} \leq q(x_2)_2, p_{(3)} \leq q(x_3)_3] \\
\approx& P[E_1 \leq mq(x_1)_1, E_1 + E_2 \leq mq(x_2)_2, E_1 + E_2 + E_3 \leq mq(x_3)_3] \\
=& \int_0^{mq(x_1)_1} \int_0^{mq(x_2)_2 - x} \int_0^{mq(x_3)_3 - x - y} e^{-x-y-z} dz dy dx \\
=& 1 - e^{-mq(x_1)_1} - mq(x_1)_1 e^{-mq(x_2)_2} - m^2 q(x_1)_1 q(x_2)_2 e^{-mq(x_3)_3} + (mq(x_1)_1)^2 e^{-mq(x_3)_3}/2.
\end{aligned}
$$

Here $q(x_1)_1$ is the x_1 quantile of $Beta(1, m)$, $q(x_2)_2$ is the x_2 quantile of $Beta(1, m-1)$, and $q(x_3)_3$ is the x_3 quantile of $Beta(1, m-2)$.

We can look at the distributions of $\tilde{p}_{(k)}$ under the alternative where $M_0 = m_0$ of the genes are from H_0 and the other $M_1 = m - M_0$ genes are from H_1. Here M_0 is random as in the preceding chapter. Again m can be observed but m_0 will be

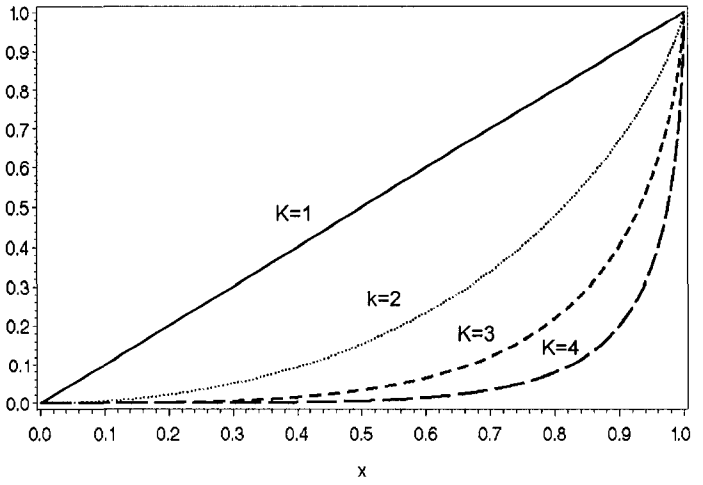

Figure 9: The estimated distribution function $P[\tilde{p}_{(k)} \leq x]$ for $m = 10000$, $k = 1, 2, 3, 4$, using equation (5.2.1).

unknown. As before we assume $M_0 \sim Binom(m, \lambda)$ as $m \to \infty$, and λ is assumed to be a known constant. We know that the vector $(mp_{(1)}, \cdots, mp_{(k)})$ has an asymptotic distribution given by the inter-point distance of a Poisson process with intensity function $g = \lambda + (1 - \lambda) \cdot d_\delta(0)$ ($d_\delta(x)$ is the density function of the p-values under H_1). The asymptotic marginal and joint distributions of $\tilde{p}_{(1)}, \tilde{p}_{(2)}, \cdots, \tilde{p}_{(k)}$ will be similar to the expressions under H_0^c, with the difference that E_i are independent exponential variables of mean $1/g$. That is,

$$P[\tilde{p}_{(k)} \leq x] \approx P[\tilde{p}_{(k-1)} \leq x] - g^{k-1} e^{-mq(x)_k g} \int_0^{mq(x)_1} \cdots \int_0^{mq(x)_k - x_1 - \cdots - x_{k-2}} 1 dx_1 \cdots dx_{k-1}. \tag{5.2.2}$$

As we mentioned before, the Poisson approximation of the mixed distribution is only good for moderate values of δ and in general, it is safe to use the Poisson approximation for $g < 1.5$.

Table 27 illustrates the relationship between the sample size (number of slides) and the number of rejections for p-values from a mixed distribution. It is assumed that for the $m = 10000$ p-values, $M_0 \sim Binom(m, 0.99)$ of them are from U(0,1) and the rest

		b=4	b=8	b=10	b=15
a=1		$\delta = 1$	$\delta = 2$	$\delta = 2.5$	$\delta = 3.75$
		g=1.02	g=1.14	g=1.35	g=4.30
$P[\tilde{p}_{(k)} \leq 0.01]$	k=1	0.01016(0.0096)	0.0114(0.0106)	0.0135(0.0124)	0.0423(0.0246)
	k=2	0.00005 (0)	0.00007(0)	0.00009(0)	0.0009(0.0003)
	k=3	1.8E-7(0)	2.5E-7(0)	4.2E-7(0)	0.00001(0)
	k=4	4.5E-10(0)	7.1E-10(0)	1E-9(0)	
$P[\tilde{p}_{(k)} \leq 0.05]$	k=1	0.0508(0.0519)	0.0568(0.0525)	0.0671(0.0659)	0.1979(0.1055)
	k=2	0.0013(0.002)	0.0016(0.0021)	0.0023(0.0019)	0.0210(0.0070)
	k=3	0.00002(0)	0.00003(0)	0.00005(0)	0.0015(0.0002)
	k=4	3.0E-7(0)	4.7E-7(0)	9.2E-7(0)	0.00008(0)
$P[\tilde{p}_{(k)} \leq 0.1]$	k=1	0.1015(0.1036)	0.1132(0.1085)	0.1329(0.1274)	0.3642(0.2028)
	k=2	0.0053(0.0054)	0.0067(0.0060)	0.0093(0.0078)	0.0763(0.0226)
	k=3	0.0002(0.0002)	0.0003(0.0001)	0.0004(0.0002)	0.0111(0.0019)
	k=4	5.0E-6(0)	7.9E-6(0)	0.00002(0)	0.0012(0.0001)
$P[\tilde{p}_{(k)} \leq 0.2]$	k=1	0.2029(0.2069)	0.2246(0.2253)	0.2607(0.2434)	0.6168(0.3661)
	k=2	0.0221(0.0207)	0.0274(0.0264)	0.0374(0.0307)	0.2493(0.0798)
	k=3	0.0016(0.0015)	0.0023(0.0024)	0.0037(0.0034)	0.0730(0.0121)
	k=4	0.00009(0)	0.0001(0.0001)	0.0003(0.0001)	0.0166(0.0009)
a=2		$\delta = 4$	$\delta = 8$	$\delta = 10$	$\delta = 15$
		g=1.08	g=3.05	g=10.97	g=515
$P[\tilde{p}_{(k)} \leq 0.01]$	k=1	0.0108(0.012)	0.0302(0.0285)	0.1044(0.0746)	0.9943(0.5855)
	k=2	0.00006(0)	0.0005(0.0006)	0.0057(0.0032)	0.9650(0.2215)
	k=3	2.1E-7(0)	5E-6(0)	0.0002(0)	0.8893(0.0573)
	k=4	5.7E-10(0)	3.6E-8(0)	0.00001(0)	0.7587(0.0128)
$P[\tilde{p}_{(k)} \leq 0.05]$	k=1	0.0538(0.0539)	0.1448(0.1235)	0.4302(0.2842)	1(0.9375)
	k=2	0.0015(0.0018)	0.0110(0.0090)	0.1097(0.0457)	1(0.7624)
	k=3	0.00003(0)	0.0006(0.0004)	0.0196(0.0058)	1(0.5286)
	k=4	3.7E-7(0)	0.00002(0)	0.0027(0.0007)	1(0.3121)
$P[\tilde{p}_{(k)} \leq 0.1]$	k=1	0.1073(0.1038)	0.2748(0.2369)	0.6851(0.4704)	1(0.9910)
	k=2	0.0060(0.0065)	0.0418(0.0323)	0.3212(0.1331)	1(0.9383)
	k=3	0.0002(0.0004)	0.0044(0.0038)	0.1110(0.0284)	1(0.8257)
	k=4	6.3E-6(0)	0.0003(0.0001)	0.0301(0.0045)	1(0.6578)
$P[\tilde{p}_{(k)} \leq 0.2]$	k=1	0.2138(0.2121)	0.4936(0.4229)	0.9134(0.7075)	1(0.9993)
	k=2	0.0247(0.0235)	0.1491(0.1062)	0.7017(0.3481)	1(0.9929)
	k=3	0.0019(0.0019)	0.0318(0.0201)	0.4426(0.1269)	1(0.9743)
	k=4	0.0001(0.0001)	0.0052(0.0017)	0.2313(0.0036)	1(0.9279)

Table 27: Estimated $P[\tilde{p}_{(k)} \leq x]$ with $x = 0.01, 0.05, 0.1$, and 0.2, for $t = 2$, $m = 10000$, $\lambda = 0.99$, $a = 1, 2$, and different b. It is assumed that $m_0 \sim bino(m, \lambda)$ genes are from the null distribution and the remaining $m - m_0$ are from the alternative distribution $H_a : K'T = a1$. Equation(5.2.2) was used with exponential variables $E_i \sim Exp(1/g)$. Here g is the mean of the Poisson approximation for the mixed distribution. The numbers in the bracket are the probabilities estimated from 10,000 simulations for the BIB design.

$m-M_0$ are from the alternative distributions $D_\delta(x)$ (calculated under $H_a : K'T = a1$). Here $t = 2$ treatments were considered with $a = 1, 2$. The probability of rejecting at least k p-values $P[\tilde{p}_{(k)} \leq x]$ is estimated from equation (5.2.2). It confirms that this method is conservative: for $a = 1$, the probability of rejecting at least one p-value at level 0.2 is about 0.2607 with $b = 10$. For settings with big g, we still listed their estimated probability here and we put the probability estimated from ten thousand simulations for the BIB design in the bracket for comparison. It shows that the approximation of equation (5.2.2) is reasonably good for even $g = 3$ in this case.

5.3 Discussion of an Alternative Adjusting Method

Sidak's method is still conservative and it only uses the information among $m - k$ p-values in the second step. An anonymous reader of a preliminary version of this thesis suggested that we consider all m p-values and use $\mathbb{P}_{(k)}$ instead. That is, define the k^{th} p-value $p_{(k)}$ as

$$\tilde{\tilde{p}}_{(k)} = max(\tilde{\tilde{p}}_{(k-1)}, P(\mathbb{P}_{(k)} \leq p_{(k)} | H_0^c)). \tag{5.3.1}$$

Again, under the independence assumption and H_0^c, all p-values are iid U[0,1]. The second term above is

$$P(\mathbb{P}_{(k)} \leq p_{(k)}) = \int_0^{p_{(k)}} \frac{m!}{(k-1)!(m-k)!} x^{k-1}(1-x)^{m-k} dx$$

which is the cdf of a beta distribution with parameters k and $m - k + 1$ at point $p_{(k)}$. That is, $P[Beta(k, m - k + 1) \leq p_{(k)}]$. Since it is the probability that at least k of the m independent uniformly distributed random variables are less than $p_{(k)}$, it is a probability given by summing over the binomial distribution with success rate $p_{(k)}$. That is, $P[Beta(k, m - k + 1) \leq p_{(k)}] = P[Bino(m, p_{(k)}) \geq k]$. Thus, we have

$$\begin{aligned} P(\mathbb{P}_{(k)} \leq p_{(k)}) &= P[Bino(m, p_{(k)}) \geq k] \\ &= 1 - \sum_{i=0}^{k-1} \binom{m}{i} p_{(k)}^i (1 - p_{(k)})^{m-i} \end{aligned}$$

As before, the first k modified adjusted p-values could be approximated by

$$\tilde{\tilde{p}}_{(k)} \approx max(\tilde{\tilde{p}}_{(k-1)}, 1 - \sum_{i=0}^{k-1} \binom{m}{i} (\sum_{j=1}^{k} E_j/m)^i (1 - \sum_{j=1}^{k} E_j/m)^{m-i}).$$

The original Westfall-Young adjusted p-values could be expressed similarly using Binomial variables:

$$\tilde{p}_{(k)} = max(\tilde{p}_{(k-1)}, P(min_{i \in \{r_k, \cdots, r_m\}} \mathbb{P}_{(i)} \leq p_{(k)}) | H_0^c)$$

and under H_0^c and the independence assumption, the second term is

$$\begin{aligned} P(min_{i \in \{r_k, \cdots, r_m\}} \mathbb{P}_{(i)} \leq p_{(k)}) &= \int_0^{p_{(k)}} (m-k+1)(1-x)^{m-k} dx \\ &= P[Beta(1, m-k+1) \leq p_{(k)}] \\ &= P[Bino(m-k+1, p_{(k)}) \geq 1] \end{aligned}$$

Numeric calculations suggest that $P[Bino(m, p_{(k)}) \geq k]$ is always smaller than $P[Bino(m-k+1, p_{(k)}) \geq 1]$ for $k > 1$, and the difference is largest in a range of values of $p_{(k)}$ roughly close to $k/(2m)$ (see figure 10 as an example). Hence, the modified adjusted p-values $\tilde{\tilde{p}}_{(k)}$ are smaller than the original Westfall-Young $\tilde{p}_{(k)}$ (equal for $k = 1$) and thus, less conservative. Obviously the modified adjusted p-values $\tilde{\tilde{p}}_{(k)}$ control FWER at α.

Since

$$P[\tilde{\tilde{p}}_{(k)} \leq x] = P[max(P(Beta(1, m) \leq p_{(1)}), \cdots, P(Beta(k, m-k+1) \leq p_{(k)})) \leq x],$$

defining $q(x)_i$ to be the x quantile of $beta(k, m-i+1)$, using equations (5.2.1) and (5.2.2), we will be able to calculate the asymptotic distribution of the modified adjusted p-values. Tables 28 illustrates the estimated $P[\tilde{\tilde{p}}_{(k)} \leq x]$ for $m = 10000$ with all p-values from U(0,1) and table 29 illustrates the estimated probability for $M_0 \sim bino(m, \lambda)$ from U(0,1) and the remaining $m - M_0$ from the alternative distribution $D_\delta(x)$ (same setting as in table 27). Both tables confirm that the probability of rejection is larger for the modified adjusted p-values than the original step-down Sidak method.

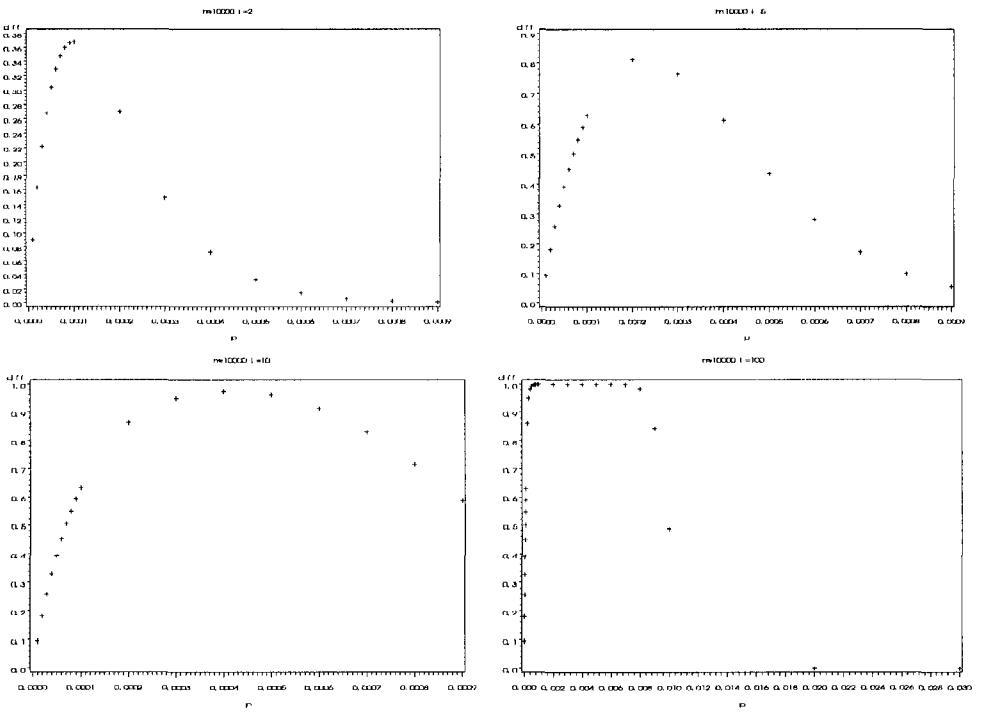

Figure 10: The difference between $P[Bino(m, p_{(k)}) \geq k]$ and $P[Bino(m-k+1, p_{(k)}) \geq 1]$ for $m = 10000$ and $i = 2, 5, 10, 100$.

	k=1	k=2	k=3	k=4
x=0.01	0.0100	0.0013	0.0004	0.0002
x=0.05	0.0500	0.0140	0.0066	0.0039
x=0.1	0.1001	0.0381	0.0213	0.0142
x=0.2	0.2000	0.1021	0.0678	0.0507

Table 28: Estimation $P[\tilde{p}_{(k)} \leq x]$ using equation (5.2.1). It is assumed that 10000 genes are tested simultaneously and null hypotheses are true for all of them.

		b=4	b=8	b=10	b=15
a=1		$\delta = 1$	$\delta = 2$	$\delta = 2.5$	$\delta = 3.75$
		g=1.02	g=1.14	g=1.35	g=4.30
$P[\tilde{\tilde{p}}_{(k)} \leq 0.01]$	k=1	0.01016(0.0096)	0.0114(0.0106)	0.0135(0.0124)	0.0423(0.0246)
	k=2	0.0014 (0.0013)	0.0017(0.0017)	0.0024(0.0012)	0.0195(0.0066)
	k=3	0.0004(0.0003)	0.0006(0.0007)	0.0009(0.0004)	0.0154(0.0032)
	k=4	0.0002(0)	0.0003(0.0004)	0.0005(0.0002)	0.0142(0.0023)
$P[\tilde{\tilde{p}}_{(k)} \leq 0.05]$	k=1	0.0508(0.0519)	0.0568(0.0525)	0.0671(0.0659)	0.1979(0.1055)
	k=2	0.0145(0.0140)	0.0178(0.0153)	0.0242(0.0192)	0.1500(0.0492)
	k=3	0.0069(0.0068)	0.0092(0.0069)	0.0139(0.0106)	0.1407(0.0339)
	k=4	0.0041(0.0032)	0.0058(0.0050)	0.0098(0.0067)	0.1384(0.0273)
$P[\tilde{\tilde{p}}_{(k)} \leq 0.1]$	k=1	0.1015(0.1036)	0.1132(0.1085)	0.1329(0.1274)	0.3642(0.2028)
	k=2	0.0392(0.0374)	0.0477(0.0430)	0.0635(0.0544)	0.3182(0.1192)
	k=3	0.0222(0.0196)	0.0290(0.0242)	0.0426(0.0328)	0.3100(0.0884)
	k=4	0.0149(0.0119)	0.0207(0.0173)	0.0331(0.0236)	0.3082(0.0751)
$P[\tilde{\tilde{p}}_{(k)} \leq 0.2]$	k=1	0.2029(0.2069)	0.2246(0.2253)	0.2607(0.2434)	0.6168(0.3661)
	k=2	0.1048(0.1075)	0.1252(0.1212)	0.1617(0.1445)	0.5891(0.2638)
	k=3	0.0703(0.0703)	0.0893(0.0824)	0.1252(0.1024)	0.5851(0.2222)
	k=4	0.0529(0.0505)	0.0709(0.0625)	0.1064(0.0825)	0.5844(0.1988)
a=2		$\delta = 4$	$\delta = 8$	$\delta = 10$	$\delta = 15$
		g=1.08	g=3.05	g=10.97	g=515
$P[\tilde{\tilde{p}}_{(k)} \leq 0.01]$	k=1	0.0108(0.012)	0.0302(0.0285)	0.1044(0.0746)	0.9943(0.5855)
	k=2	0.0015(0.0013)	0.0107(0.0090)	0.0827(0.0427)	0.9943(0.5827)
	k=3	0.0005(0.0006)	0.0072(0.0057)	0.0813(0.0359)	0.9943(0.5825)
	k=4	0.0002(0.0001)	0.0060(0.0038)	0.0812(0.0331)	0.9943(0.5825)
$P[\tilde{\tilde{p}}_{(k)} \leq 0.05]$	k=1	0.0538(0.0539)	0.1448(0.1235)	0.4302(0.2842)	1(0.9375)
	k=2	0.0161(0.0166)	0.0919(0.0650)	0.4188(0.2349)	1(0.9372)
	k=3	0.0080(0.0074)	0.0789(0.0493)	0.4185(0.2182)	1(0.9372)
	k=4	0.0049(0.0043)	0.0742(0.0433)	0.4185(0.2109)	1(0.9372)
$P[\tilde{\tilde{p}}_{(k)} \leq 0.1]$	k=1	0.1073(0.1038)	0.2748(0.2369)	0.6851(0.4704)	1(0.9910)
	k=2	0.0433(0.0423)	0.2113(0.1587)	0.6817(0.4227)	1(0.9910)
	k=3	0.0254(0.0231)	0.1950(0.1321)	0.6816(0.4091)	1(0.9910)
	k=4	0.0176(0.0166)	0.1894(0.1199)	0.6816(0.4025)	1(0.9910)
$P[\tilde{\tilde{p}}_{(k)} \leq 0.2]$	k=1	0.2138(0.2121)	0.4936(0.4229)	0.9134(0.7075)	1(0.9993)
	k=2	0.1148(0.1164)	0.4385(0.3400)	0.9132(0.6783)	1(0.9993)
	k=3	0.0795(0.0808)	0.4248(0.3089)	0.9132(0.6686)	1(0.9993)
	k=4	0.0616(0.0625)	0.4204(0.2919)	0.9132(0.6645)	1(0.9993)

Table 29: Estimated $P[\tilde{\tilde{p}}_{(k)} \leq x]$ with $x = 0.01, 0.05, 0.1$, and 0.2. Same setting as in table 27 except that $q(x)_i$ are the x quantile of beta(k, m-i+1).

Both the step-down Sidak adjustment and the modified adjustment will reject hypotheses if their corresponding adjusted p-values are less than or equal to α. In order to illustrate the difference between the two methods, we consider two separate situations. In the situation that all null hypotheses are true, all p-values are from $U(0,1)$ and both methods compare the ordered p-values ($min_{\{r_k,\cdots,r_m\}}\mathbb{P}_{(i)}$ and $\mathbb{P}_{(k)}$) with their corresponding observed ones $p_{(k)}$. In the second situation when some of the hypotheses are from the null and some are from the alternative, the p-values would have a mixture distribution and both methods have some shortcomings. We will illustrate heuristically the difference between these two methods through an example. Assume that 10 p-values are considered and 2 of them are from the alternative. Denote the p-values from the uniform by x_i and the p-values from the alternative by y_j. Suppose we observe $y_1 < x_1 < y_2 < x_2 < \cdots < x_8$. To calculate $\tilde{p}_{(2)}$, the Sidak method will discard the first observed p-value y_1 and compare $min_{\{2,\cdots,10\}}\mathbb{P}_{(i)}$ (which has mean 1/10) to x_1 (which has mean 1/9). For $\tilde{p}_{(4)}$, it compares $min_{\{4,\cdots,10\}}\mathbb{P}_{(i)}$ (mean 1/8) to x_2 (mean 2/9). Both times the $min\mathbb{P}_{(i)}$ has mean smaller than the mean of the x's, thus the adjusted p-values will likely be large. Therefore, this method is conservative but robust since usually p-values from the alternative have larger probabilities of falling among the first $k-1$ observed p-values than p-values from $U(0,1)$ but they are discarded in the calculation of $\tilde{p}_{(k)}$. However, the modified adjustment considers all m p-values and used $P_{(k)}$ to adjust $\tilde{\tilde{p}}_{(k)}$. That is, to calculate $\tilde{\tilde{p}}_{(2)}$, it compares $P_{(2)}$ of $m = 10$ (mean 2/11) to x_1 (mean 1/9). For $\tilde{\tilde{p}}_{(4)}$, it compares $P_{(4)}$ of $m = 10$ (mean 4/11) to x_2 (mean 2/9). Both $P_{(2)}$ and $P_{(4)}$ have means larger than the means of the x's and thus the adjusted p-values will tend to be small. The more p-values among the first $k-1$ that are from the alternative, the smaller the adjusted p-values $\tilde{\tilde{p}}_{(k)}$ will be. Therefore, the modified adjustment is less conservative than the step-down Sidak method but it is highly dependent on the positions of the p-values from the alternative. Consequently, it is not robust to p-values from the alternative: once it rejects a hypothesis, it is likely to reject many more. Thus, it could yield a large number of false rejections.

An example is given to illustrate the difference in the number of rejections between the step-down Sidak adjustment and the modified adjustment. Assume 10 p-values

number of rejections											
					$\tilde{\tilde{p}}_{(k)}$						
$\tilde{p}_{(k)}$	0	1	2	3	4	5	6	7	8	9	total
0	5606										5606
1	52	1261	566	336	248	232	174	126	123	104	3222
2	29	8	195	165	115	110	99	90	62	61	934
3	10	3	5	16	23	29	27	29	35	18	195
4	8	0	0	1	3	8	4	6	7	1	38
5	0	1	0	0	0	0	1	1	1	1	5
total	5705	1273	766	518	389	379	305	252	228	185	10000

Table 30: The number of rejections using $\tilde{p}_{(k)}$ and $\tilde{\tilde{p}}_{(k)}$. Ten thousand simulations are performed for 10 p-values with $M_0 \sim Binom(10, 0.8)$ from U(0,1) and the rest from alternative: $t = 2, b = 10, \delta = 10$.

are considered, $M_0 \sim Binom(10, 0.8)$ p-values are iid U(0,1) and the rest are from a distribution $D_\delta(x)$ with $t = 2, b = 10, a = 2$, which leads to $\delta = 10$. That is, on average, 2 p-values are from the alternative distribution where the gene is 4 fold expressed under the treatment condition. Ten thousand simulations are performed to generate the p-values and table 30 lists the number of rejections at level $\alpha = 0.05$ using the step-down Sidak method and the modified adjustment. It shows that the modified adjustment not only rejects more p-values than the step-down Sidak method, but also rejects many more than the expected number of p-values from the alternative distribution. For example, the step-down Sidak method rejects more than 2 hypotheses 238 times, while the modified adjustment rejects more than 2 hypotheses 2256 times, which includes rejecting 9 out of 10 hypotheses 185 times. That is, the method leads to a large number of false rejections, and so cannot be recommended.

5.4 Example Data Set

The step-down Sidak method and the modified adjusted p-values 5.3.1 were applied to the HIV data set illustrated in Table 1. Details will be discussed in the next

i	gene	raw $p_{(k)}$	Sidak $\tilde{p}_{(k)}$	Modified $\tilde{\tilde{p}}_{(k)}$	i	gene	raw $p_{(k)}$	Sidak $\tilde{p}_{(k)}$	Modified $\tilde{\tilde{p}}_{(k)}$
1	R97915	0.000002	0.0135	0.0135	6	R20409	0.000015	0.1321	0.0135
2	R69570	0.000002	0.0152	0.0135	7	H12862	0.000022	0.1848	0.0135
3	R60223	0.000003	0.0270	0.0135	8	H59825	0.000024	0.1967	0.0135
4	R78620	0.000007	0.0628	0.0135	9	T82048	0.00004	0.3058	0.0135
5	R99977	0.000010	0.0829	0.0135	10	R31076	0.00004	0.3062	0.0135

Table 31: The first 10 raw and adjusted p-values of the HIV data.

chapter. Table 31 lists the first 10 raw and adjusted p-values. The step-down Sidak method rejects 3 null hypotheses with controlling FWER at 0.05 while the modified adjusted p-values $\tilde{\tilde{p}}_{(k)}$ rejects more than 8900 null hypotheses. Clearly, the step-down Sidak method is likely too conservative and leads to a large number of type II errors, while the modified method leads to too many type I errors and should not be used as a multiple testing procedure.

Chapter 6

Application

6.1 Example Data

A data set collected from a HIV experiment was used in the previous two chapters to illustrate the applications of these different multiple testing methods. We thank Dr. Diaz-Mitoma and Dr. Ikuri Alvarez-Maya from the Children's Hospital of Eastern Ontario for supplying this data. The experiment compared the genetic expression of Peripheral Blood Mononuclear Cells (PBMC) from HIV positive humans to genetic expressions of PBMC from HIV positive monkeys. The experiment was divided into two parts where each part contains half of the 19,000 genes. We will examine only the first part here. Three slides were produced for the genes in the first part of this experiment. In the first slide, human samples were dyed in green and Monkey samples were dyed in red, and in the other two slides, the color was flipped - human samples in red and monkey samples in green. Each slide contains 9358 genes: 9132 of them are attached on two spots, and 226 genes are attached on 4 or 6 spots. There are also some housekeeping genes and we exclude them from the data analysis. Intensities in two spots on slide 1 and one spot on slide 3 were measured twice (reason unknown), both measures are excluded from the analysis. The genes with different numbers of replicates will cause the degrees of freedom of the F-test statistics that will be used in the data analysis to be different. Although their corresponding p-values will still have uniform distributions under the null hypotheses (and thus the analysis method

will be same), to be consistent with our assumptions in the corollaries in Appendix A, we will apply our analysis only on the 9132 genes which have the same number of replicates for illustration.

The intensity and background on every spot of every slide for each color are measured. Both intensities and background are log transformed with base 2. The contour plots of log transformed backgrounds and intensities indicate that background correction is necessary. Figure 11 illustrates the contour plots of the monkey sample from slide 1. The top left plot shows that the background measures at small x and y locations are high and the measures at large x and y locations are low. The top right plot of intensities gives a similar message. Therefore, background correction is necessary and we corrected the log transformed intensities by subtracting the log transformed background measure from it. The bottom plot of Figure 11 gives the background corrected intensities. It shows that the intensities after background correction are randomly distributed through the x and y locations and no specific pattern exists. The contour plots of other slides of both colors behave similarly. Consequently, background correction is performed on intensities on both colors and on all 3 slides.

The color bias is then corrected through applying LOESS (Locally weighted polynomial regression) fit with $M = log_2$(intensities of Human sample) $- log_2$(intensities of Monkey sample) on $A = log_2$(intensities of Human sample) $+ log_2$(intensities of Monkey sample) for each slide. After that the slide normalization is performed: the medians are all set to 0 and the inter-quartile ranges are set to 1 for each slide. Now the data from the 3 slides are combined for analysis. For each gene the model $y_{ijk} = \mu + B_i + T_j + e_{ijk}$ is applied. The y_{ijk} is the log transformed intensities on spot k of slide i for sample j, μ represents the overall mean, B_i is the effect of slide i on intensities, T_j is the effect of sample j (j=1 for human sample and j=2 for monkey sample), and e_{ijk} is an error term. Here dye effect is not considered since the color bias is corrected already.

To test the null hypothesis H_0: a gene is not differentially expressed, the test statistic and the corresponding p-value for each gene is calculated. Both multiple test methods - FDR and Westfall-Young adjusted p-values - are applied on the p-values to detect the differentially expressed genes. Table 32 list the first 20 smallest

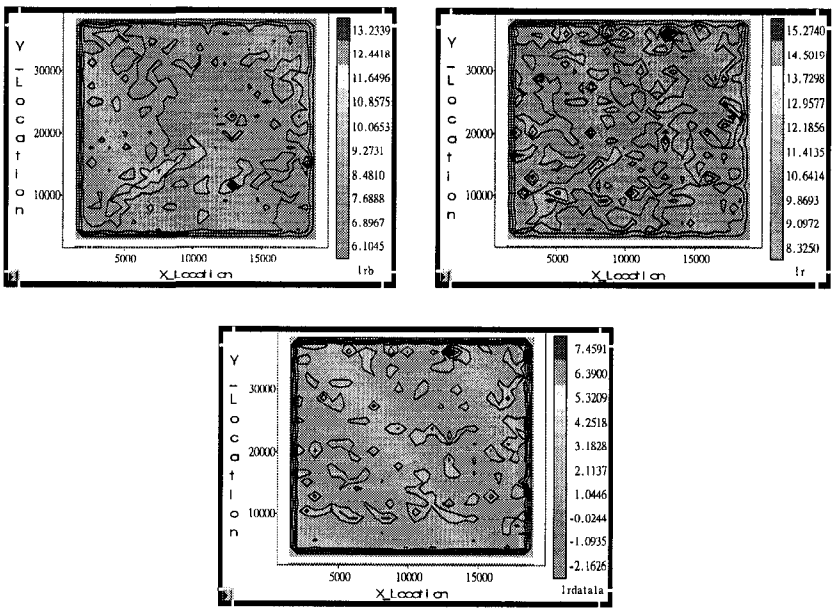

Figure 11: The contour plots of log transformed background (top left), log transformed intensities of the monkey sample from slide 1 (top right), and log transformed intensities after background correction (log(intensity)-log(background)) (bottom).

unadjusted p-values, their corresponding FDR threshold for $q* = 0.05$, adjusted p-values using the step-down Sidak method $\tilde{p}_{(i)}$, and modified adjusted p-values $\tilde{\tilde{p}}_{(i)}$. The FDR threshold shows that the FDR method will reject 14 null hypotheses, which is a relatively large number of rejections in a microarray experiment with 6 replicates for each gene. The step down Sidak method rejects 3 null hypotheses with controlling FWER at 0.05, and the modified method rejects 8929 null hypotheses with controlling FWER at 0.05 since the first 8786 $\tilde{\tilde{p}}_{(i)}$ all equal 0.0135. The result clearly indicates that the modified method may lead to a large number of false rejections. Even under the assumption of independence, we have seen that the modified method can lead to an increased number of false rejections. This problem may be exacerbated when the genes are highly correlated.

6.2 Example of Planning an Experiment

We will illustrate an application of the FDR and adjusted p-values methods on planning an Affymetrix experiment involving the mobility of rats.

A preliminary experiment was done by Dabrowski et.al [9] on 4 rats to obtain some prior information such as the error variance. Two rats were represented twice in the experiment, and the data were treated as having come from 6 separate rats. In this planning step, this simplification of the analysis was not a significant restriction, particularly in view of the limited number of subjects in the trial. In total 31042 genes were measured on each rat. The raw signal readings were integers from 0 to several thousand and they were transformed by the function $X \rightarrow log_2(1+X)$. The readings from each slide were normalized by applying linear adjustment to have medians 0 and upper quartile point 1. Then the normalized data were collected into a single data set with 31042 rows, with each row corresponding to a single probeset. Each probset had 6 readings corresponding to the 6 slides of this experiment. The original datasets contained additional reading at additional Affymetrix probesets, but these are not retained since they are intended for quality control purposes only. The error standard deviation (i.e. the standard deviation of biological+measurement error) was estimated from the individual sample standard deviations computed for each of the

k	gene	p-values	FDR	$\tilde{p}$	$\tilde{\tilde{p}}$
1	R97915	0.000002	0.000005	0.0135	0.0135
2	R69570	0.000002	0.000011	0.0152	0.0135
3	R60223	0.000003	0.000016	0.0270	0.0135
4	R78620	0.000007	0.000022	0.0628	0.0135
5	R99977	0.000010	0.000028	0.0829	0.0135
6	R20409	0.000015	0.000033	0.1321	0.0135
7	H12862	0.000022	0.000038	0.1848	0.0135
8	H59825	0.000024	0.000044	0.1967	0.0135
9	T82048	0.00004	0.000049	0.3058	0.0135
10	R31076	0.00004	0.000055	0.3062	0.0135
11	H10026	0.00004	0.000060	0.3083	0.0135
12	R79387	0.00005	0.000066	0.3681	0.0135
13	H71213	0.00006	0.000071	0.4341	0.0135
14	H82521	0.00006	0.000077	0.4448	0.0135
15	T93912	0.00016	0.000082	0.7578	0.0135
16	H45241	0.00017	0.000088	0.7960	0.0135
17	H17394	0.00018	0.000093	0.7979	0.0135
18	R06552	0.00018	0.000099	0.8037	0.0135
19	H82992	0.00018	0.00010	0.8065	0.0135
20	H29063	0.00018	0.00011	0.8099	0.0135

Table 32: The first 20 smallest unadjusted p-values, with corresponding FDR threshold for $q* = 0.05$, adjusted p-values by step-down Sidak method $\tilde{p}$, and by the modified adjusted p-values $\tilde{\tilde{p}}$.

31042 probesets. The vast majority of the estimated standard deviations $\hat{\sigma}$ are at most 1, and among probesets with a higher anticipated mean response, $\hat{\sigma}$ is generally bounded above by 0.5.

Dabrowski et.al [9] designed an experiment for a single gene to detect changes in gene expression due to mobility or immobility either overall or somewhere over 5 time points. A two factor analysis of variance (ANOVA) is used: Factor A has two levels; 0-mobile and 1-immobile, and factor B has 5 levels corresponding to measurements taken at 2, 4, 8, 16 and 32 weeks. The corresponding statistical model is:

$$Y_{ijk} = \mu + a_i + b_j + (ab)_{ij} + e_{ijk}.$$

Here Y_{ijk} represents the k^{th} rat in the treatment cell combining level i of factor A and level j of factor B. The μ represents the overall mean response across all rats, a the deviation due to the level factor A, b the deviation due to the level of B, (ab) the interaction specific to having both level i of A and level j of B, and e_{ijk} is the random (biological+measurement) error for that specific measurement. There is one rat for each observation, l rats for each pair of A and B factor level combinations, and $10l$ rats in total for the experiment. The null hypothesis is

$$H_0\text{: no A or AB effect,}$$

and the appropriate F-test is then

$$\text{Reject } H_0 : \text{ no A or AB effect}$$
$$\text{if } F = ((SS_A + SS_{AB})/5)/MSE > F(\alpha; 5, 10(l-1)) \ .$$

Alpha is the desired significance level (Type I error probability) for the test, and the test statistic distribution is F with 5 and 10(l-1) degrees of freedom under H_0. If H_0 is not true and A or AB effect exists, the test statistic distribution is non-central F with 5 and 10(l-1) degrees of freedom and non-centrality parameter $\delta = \delta_A + \delta_{AB}$, with

$$\delta_A = (5 \sum_{i=1}^{2} (a_i)^2)/\sigma^2$$
$$\delta_{AB} = (l \sum_{i=1}^{2} \sum_{j=1}^{5} (ab)_{ij}^2)/\sigma^2.$$

Since the log-transformation is used on the intensity, a doubling of expression corresponds to a shift of the mean of the log-transformed data by 1. Dabrowski et.al [9] illustrated by applying the Bonferroni adjustment for 10 specific probesets, the sample size $10l$ needed to achieve a solid probability of detecting a doubling in the probeset expression if it is present.

We will employ the FDR and the adjusted p-values $\tilde{p}_{(k)}$ on the 31042 p-values to calculate the probability of detecting at least k genes with replications $10l$, $l = 1, \cdots, 10$, controlling FDR or FWER at 0.1. We assume on average 100 genes come from the alternative distribution - with doubling in the probeset expression and all other genes from H_0: no A or AB effects. Thus, $\lambda = 1 - 100/31042 = 0.997$. We will assume that observations across probesets are statistically independent, so that F statistics (and consequently the p-values they generate) obtained from each probeset are independent. A doubling of mean expression (independent of time) could be represented in this formula by $a_1 = 0.5 = -a_2$, $a_1 = 1, a_2 = 0$, etc. We will use the first one here for illustration, then $\delta_A = 2.5/\sigma^2$. Similarly, if there is just one time point with a doubling of gene expression, then $\delta_{AB} = 0.5l/\sigma^2$. Table 33 lists the non-centrality parameter δ for $\sigma^2 = 1$ and number of rats $10l$, $l = 1, \cdots, 10$, and the corresponding $g = \lambda + (1 - \lambda) * d_\delta(0)$. It shows that even with 100 rats, the non-centrality parameter is only 7.5. Table 34 lists the estimated lower bound of $P[R \geq k]$ by applying the FDR method. It shows that even using 100 rats, $P[R \geq 1]$ is only 0.33. Table 35 lists the $P[R \geq k]$ using the step-down Sidak method $\tilde{p}_{(k)}$. Ten thousand simulations were performed. Since the g is large for $N >= 40$, we simulated all 31042 p-values, calculated the ordered p-values, and then made corresponding adjustment. The results show that the probability $P[R \geq 1]$ is still very small (0.3193), which corresponds to a number of rats at around 100. Note that this experiment compares two treatments at five time points, thus $n = 100$ is not large relative to the number of comparisons under consideration.

number of rats	δ	$g = \lambda + (1-\lambda) * d_\delta(0)$
20	3.5	1.028
30	4	1.243
40	4.5	2.857
50	5	14.764
60	5.5	102
70	6	744
80	6.5	5471
90	7	40385
100	7.5	299204

Table 33: The non-centrality parameter δ for $\sigma^2 = 1$, $m = 31042$, $\lambda = 0.997$, and $l = 1, \cdots, 10$, and the corresponding $g = \lambda + (1-\lambda) * d_\delta(0)$. The $d_\delta(0)$ is estimated by $d_\delta(x)$ with $F_0^{-1}(1-x) = 10000$.

	number of rats N				
$P[R \geq k]$	N=20	N=30	N=40	N=50	N=60
k=1	0.1022	0.1085	0.1194	0.1358	0.1587
k=2	0.0193	0.0215	0.0253	0.0313	0.0405
k=3	0.0041	0.0047	0.0059	0.0078	0.0110
k=4	0.0009	0.0011	0.0014	0.0020	0.0031
$P[R \geq k]$	N=70	N=80	N=90	N=100	
k=1	0.1892	0.2280	0.2759	0.3330	
k=2	0.0541	0.0739	0.1017	0.1397	
k=3	0.0163	0.0247	0.0381	0.0587	
k=4	0.0049	0.0083	0.0142	0.0243	

Table 34: The lower bound of probability of detecting at least k genes $P[R \geq k]$ calculated by the FDR method, with 100 genes which have doubling in the probeset expression and the other genes from H_0: no A or AB effect. Here $q^* = 0.1$ and $\sigma = 1$.

	number of rats N		
$P[R \geq k]$	N=40	N=50	N=100
	Step-down Sidak adjustment $\tilde{p}_{(k)}$		
k=1	0.1185	0.1277	0.3193
k=2	0.0068	0.0080	0.0629
k=3	0.0003	0.0006	0.0084
k=4	0.0001	0.0001	0.0008

Table 35: The probability of detecting at least k genes $P[R \geq k]$ calculated by the step-down Sidak method, with 100 genes which have doubling in the probeset expression and the other genes from H_0: no A or AB effect. A hypothesis is rejected when its adjusted p-values is smaller than 0.1. Ten thousand simulations were performed. Adjustment were made on ordered p-values calculated from simulated all 31042 p-values.

Chapter 7

Conclusion and Discussion

In cDNA microarray experiments, each slide compares gene sequences from two samples (treatments) under study. When the number of treatments is more than 2, how to arrange the treatment groups on slides becomes an issue. Different designs could be applied and this thesis compared the power of three different designs to detect a single differentially expressed gene. A two way model is used with the slide and the treatment group as the factors. It is assumed that the data used in this model are already normalized so that the colour bias is removed. Kerr, et al [21] focused on data analysis through applying a model with more factors, and compare two treatments each time. Here, we focus more on experimental design, in terms of comparing different designs and determining the sample sizes, so we used a simplified model. An F statistic is used to test a matrix of treatment contrasts: $H_0 : T_i - T_t = 0, for\ \ i = 1, \cdots, t-1$, vs $H_a : T_i - T_t = a_i$. It is concluded that using the same numbers of slides to compare the same number of treatments t, the BIB design is always better than the reference design. There is no absolute conclusion regarding which one of the BIB or loop designs is better for $t > 3$. In general, the loop design is better than the BIB design when the treatment differences a_i are different and the treatments with large effects are distributed evenly in the loop. If the treatments with large effects are clustered in the loop, the loop design is worse than the BIB design. The loop design is better than the reference design when the number of treatments to be compared t is less than 5. There is no absolute conclusion that one of the loop or reference designs is better for

$t > 5$. For a specific number of treatments, we calculated formulas to compare the designs. The exact density function of the p-value of the F test statistic under H_a is calculated analytically and an approximation of the density at 0 is developed for later use.

Different models could be applied in the same manner when necessary, such as a model with dye effects added, a model with interaction terms added, or a mixed model with some terms treated as random. However, these models make it more complicated to calculate the non-centrality parameters of the test statistics, which is the key used to compare the three designs in terms of power.

Multiple testing procedures are then applied to the p-values obtained from the F test statistics to determine which genes are differentially expressed. This thesis considered the FDR method and the Westfall-Young resampling method under the assumption that all single gene tests are independent. The FDR method is very easy to apply and the expected false rejection rate is controlled at q^*. We calculated lower bounds for the probability that at least k genes are detected $P[R \geq k]$ and for the probability that given M_1 truly differentially expressed genes, j of them are detected $P[S > j|M_1 = m_1]$, assuming that the percentage of non-differentially expressed genes goes to a constant as the number of genes goes to infinity and the test statistics of the truly differentially expressed genes have an F distribution with a common fixed non-centrality parameter. The second assumption is not very practical, but it is not a severe restriction in terms of experimental design. Even if genes have different non-centrality parameters, if they could be grouped into several classes such that each class has a common non-centrality parameter, the method of calculating the lower bounds will be similar. Our motivation to focus on the probability $P[R \geq k]$ is that it would be helpful to have some knowledge of the chance of rejecting a moderate number of incorrect null hypotheses before applying the FDR method, especially if the chance is very small or very big. The probability $P[S > j|M_1 = m_1]$ is treated as a notion of power in multiple testing and thus is of interest. These probabilities could also be used as references to determine the sample size of replications for each gene in the experimental planning stage. We also formulated a conditional probability - the probability of the number of genes that are truly differentially expressed given k

genes are detected $P[S = j|R = k]$, which turns out to have a binomial distribution with the chance of success depending on the density of p-values at point 0 under H_0.

Under an independence assumption of single gene tests, the Westfall-Young minP adjusted p-values is actually the step-down Sidak method. For the first k adjusted p-values, we applied the results from extremal processes and proposed a way to speed up the simulation which is involved in applying the step-down Sidak method in experimental designs. We also evaluated the marginal asymptotic distributions of the first k adjusted p-values. A modification on the adjustment of p-values is also discussed, which is less conservative but can lead to rejecting too many true hypotheses.

Under the independence assumption, both the FDR method and the step-down Sidak adjusted p-values are still conservative. Experimental designs without the assumption of independence or which use other multiple testing procedures merit further study.

The simplified lower bound of the FDR method using the Poisson approximation and the asymptotic distributions in the Westfall-Young adjusted p-values using exponential variables are still valid for p-values of other test statistics, given that the densities of the p-values under H_a have derivatives at point 0. The density may not be close to linear in a small neighborhood of 0, but the number of p-values that fall into a small neighborhood of 0 will still have an asymptotic Poisson distribution, although it is not homogeneous anymore and the calculation in the approximations will be complicated.

Appendix A

Weak Convergence of Extremal Processes

Corollary 4.19 in Resnick [28] establishes the weak convergence of extremal point processes. We will apply this theorem to estimate the joint distribution of the extreme p-values appearing in the FDR and the other methods listed previously. The asymptotics of these p-values are given by Resnick's result. We first quote the result in a form suited to our needs and then state our own versions.

Corollary A.0.1 (Corollary 4.19, [28]) *Let $\{X_n, n \geq 1\}$ be iid with continuous strictly increasing distribution F on $[0, \infty]$. Assume there exists a distribution function G, and for $M_n = max(X_1, \cdots, X_n)$, there exist $a_n > 0$, $b_n \in R$, $n > 1$ such that*

$$P[a_n^{-1}(M_n - b_n) \leq x] = F^n(a_n x + b_n) \to G(x). \tag{A.0.1}$$

If

$$G(x) = \begin{cases} exp\{-(-x)^\alpha\}, & if\ x < 0, \\ 1, & if\ x \geq 0 \end{cases}$$

$\alpha > 0$, then $x_0 = sup\{x : F(x) < 1\} < \infty$. Set $E = (-\infty, 0]$, $\nu(x, 0] = (-x)^\alpha$, $x < 0$, and equation (A.0.1) is equivalent to:

$$\xi_n := \sum_{i=1}^{\infty} I(\frac{i}{n}, \frac{X_i - x_0}{x_0 - \gamma_n}) \Rightarrow \xi = PRM(dt \times d\nu),$$

in $M_p([0,\infty) \times (-\infty, 0])$, *where* $\gamma_n = (1/(1-F))^{-1}(n)$.

Here "$\Rightarrow$" denotes weak convergence in the vague topology on the space of counting measure, $M_p(E)$ where E is a complete separable metric space, and $PRM(d\mu)$ denotes a Poisson random measure on E with mean measure μ. (Refer to Resnick [28]).

This result establishes the weak convergence of extremal point processes ξ_n to a planar Poisson process ξ with first coordinate equally spaced and with mean measure determined by the upper tail behaviour of the distribution function of the iid X_i. Of course, a completely parallel result holds for lower extreme values. We apply this corollary to two special cases and so develop the asymptotic behaviour of minimal p-values under the null and alternative hypotheses of Chapter 3.

Corollary A.0.2 *Denote by* $\{p_1, p_2, p_3 \cdots\}$ *the p-values of iid F-statistics generated by variables with central F distributions, i.e.* $p_i = 1 - F_0(X_i)$ *for* X_i *having the central F distribution,* F_0. *Then*

- *Let* $N_n(a) = \sum_{i=1}^{\infty} I(p_i \leq \frac{a}{n})$, *for* $a \geq 0$. *As* $n \to \infty$, $N_n \Rightarrow PRM(dt)$ *in* $M_p([0,\infty))$. *Consequently, for the first k ordered p-values* $p_{(i)}$, *where k is a finite integer relatively small compared to n, we have that* $n(p_{(i)} - p_{(i-1)})$ *are asymptotically independent exponential variables of mean 1 as* $n \to \infty$.

Proof.

As stated in Chapter 3, the p-value p_i of a central F distributed statistic is uniformly distributed on $[0,1]$. Thus, $-p_i \sim Uniform[-1,0]$. Let $a_n = \frac{1}{n}$, $b_n = 0$, then we have for $M_n = max(-p_1, \cdots, -p_n)$, with $x < 0$,

$$\begin{aligned} P(M_n \leq a_n x + b_n) &= [1 - F(-(a_n x + b_n))]^n \\ &= [1 - F(-x/n)]^n \\ &= (1 + \frac{x}{n})^n \\ &\to e^x, \end{aligned} \tag{A.0.2}$$

as $n \to \infty$. So the condition A.0.1 is satisfied, with $exp\{-(-x)^\alpha\} = G(x)$ and $\alpha = 1$ in item (iii) of Corollary A.0.1. Here, $x_0 = sup\{x : F_{-p}(x) < 1\} = 0$. Let $\gamma_n = -1/n$,

then $\frac{X_i - x_0}{x_0 - \gamma_n} = \frac{-p_i}{1/n} = -np_i$. Therefore we obtain, as $n \to \infty$, that the sample point process converges weakly to a limiting standard Poisson process. In particular,

$$\xi_n(([0,1] \times [-a,0)), \{(\tfrac{i}{n}, -np_i), i \leq n\}) \to Poisson(a),$$

i.e. $\lambda(([0,1] \times [-a,0)) = \int_0^1 \int_{-a}^0 dtdx = a$, which proves the statement: The number of $np_i \leq a$ for $i = 1, \cdots, n$ has an asymptotic Poisson distribution with mean a as $n \to \infty$. Furthermore, considering np_i as an arrival, and the intervals between successive points (arrivals) as waiting times, we know that for the first k (a finite number) waiting times between arrivals - $np_{(1)}$, $np_{(2)} - np_{(1)}$, $\cdots$, $np_{(k)} - np_{(k-1)}$, their joint distribution can be described as a function of the joint distribution of the first k arrivals. That is, the following two events are equivalent: $(np_{(k)} > t) = (N_n(t) < k)$, where $N_n(t)$ is the number of arrivals before time t (number of $np_i < t$). As $n \to \infty$, the process of arrivals is asymptotically a unit rate Poisson process, and so the waiting times have asymptotically independent exponential distributions with mean 1. That is, $(np_{(1)}, n(p_{(2)} - p_{(1)}), \cdots, n(p_{(i)} - p_{(i-1)})) \to (E_1, \cdots, E_k)$ in distribution where $E_1, \cdots, E_k$ are *i.i.d* exponential random variables with mean 1. □

Corollary A.0.3 *Suppose $\{p_1, p_2, p_3, \cdots\}$ are p-values of iid F-statistics determined by F distributions with non-centrality parameter δ, i.e. $p_i = 1 - F_0(X_i)$ for X_i having the non-central F distribution of non-centrality δ, F_δ. Denote $D_\delta(p) = \int_{F_0^{-1}(1-p)}^{\infty} f(t,\delta)dt$, and $d_\delta(0)$ be the right-hand derivative of function D_δ at the point $p = 0$.*

- *Let $N_n(a) = \sum_{i=1}^{\infty} I(p_i \leq \frac{a}{n})$, for $a \geq 0$. As $n \to \infty$, $N_n \Rightarrow PRM(d_\delta(0) \cdot dt)$ in $M_p([0,\infty))$. Consequently, for the first k ordered p-values $p_{(i)}$, where k is a finite integer relatively small compared to n, we have that $n(p_{(i)} - p_{(i-1)})$ are asymptotically independent exponential variables of mean $1/d_\delta(0)$ as $n \to \infty$.*

Proof.

As stated in Chapter 3, the p-value of an F statistic with non-centrality parameter δ has distribution function $D_\delta(p) = \int_{F_0^{-1}(1-p)}^{\infty} f(t,\delta)dt$. As $p \to 0$, the function $D_\delta(p)$ can be approximated by its secant line at 0. That is, $D_\delta(p) \approx d_\delta(0) \cdot p$.

Similarly as in the proof of the last Corollary, let $a_n = \frac{1}{nd_\delta(0)}$, $b_n = 0$, then we have for $M_n = max(-p_1, \cdots, -p_n)$, as $n \to \infty$, for $x < 0$,

$$\begin{aligned} P(M_n \leq a_n x + b_n) &= P(M_n \leq \frac{x}{nd_\delta(0)}) \\ &= [1 - D_\delta(\frac{-x}{nd_\delta(0)})]^n \\ &\approx [1 + \frac{d_\delta(0)x}{nd_\delta(0)}]^n \\ &\to e^x. \end{aligned}$$

So the condition in Corollary A.0.1 is satisfied. Here, let $\alpha = 1$ and $G(x) = exp\{-(-x)^\alpha\}$ for $x < 0$, which is the case of item (iii) of Corollary A.0.1. Since $x_0 = sup\{x : F_{-p}(x) < 1\} = 0$, and letting $\gamma_n = -1/(nd_\delta(0))$, then $\frac{X_i - x_0}{x_0 - \gamma_n} = -d_\delta(0) \cdot np_i$, and the corollary follows. □

The thesis will frequently consider collections of p-values where some are obtained under the null hypothesis, and some under a specific alternative hypothesis. Efron [12] considered the asymptotics of such a situation using the empirical Bayes approach. We will study the asymptotics of p-values using the extremal point process limit in a very similar situation. We will assume that each p-value is generated by either the null or the specific alternative according to an independent Bernoulli process. Consequently the proportion of null to alternative p-values tends to a constant as the number of terms increases. The following corollary provides the extremal point process limit. The proof follows as above.

Corollary A.0.4 *Take $p_1, p_2, \ldots$ to be independent p-values computed either under the null hypothesis (with probability $\lambda \in (0,1)$), or in the context of Corollary A.0.3 (with probability $1 - \lambda$).*

- *Let $N_n(a) = \sum_{i=1}^{\infty} I(p_i \leq \frac{a}{n})$, for $a \geq 0$. As $n \to \infty$, $N_n \Rightarrow PRM(g \cdot dt)$ in $M_p([0,\infty))$, where $g = \lambda + (1-\lambda)d_\delta(0)$. Consequently, for the first k ordered p-values $p_{(i)}$, where k is a finite integer relatively small compared to n, we have that $n(p_{(i)} - p_{(i-1)})$ are asymptotically independent exponential variables of mean $1/g$ as $n \to \infty$.*

Appendix B

Program code

The first part list the macros used. For some macros used only in a specific program, they are included in that program code.

- **macros.**

```
%macro delta(i=2,t=2);
data data&i;
    do a=0.5, 1, 2, 4;
        do b=&blist;
            delta1=a**2* b/(2* &t);
            delta2=a**2* b*(&t-1)/(4* &t**2);
            output;
        end;
    end;
run;
%mend;
%macro graphsetting;
goptions ftext=swiss htext=1.5;
symbol1 interpol=joint line=1 c=black ;
symbol2 interpol=joint line=2  c=black ;
symbol3 interpol=joint line=44  c=black ;
symbol4 interpol=joint line=34 c=black w=2 ;
legend1 label=none
```

```
        position=(top center inside)
        mode=share;
axis  label=NONE;
%mend;
%macro calcu_v(i=0);
    x&i.=(k+&i)*q/m;
    z&i.=finv(1-x&i.,&t-1, &b-&t+1, 0);
    v&i.=1-cdf('f', z&i., &t-1, &b-&t+1, delta);
%mend;
%macro text(s=1, e=5);
    m=&m; q=&q; lambda=&lambda;
    delta =((&a)**2) *&b/(2* &t);
    %do ind2=&s %to &e;
        %calcu_v(i=&ind2);
    %end;
%mend;
%macro prob_theory_L(q=0.1,t=4, b=24, a=1, start=1, end=5, step=1,
        macroC=est_k_L5_alt);
  %do ind5=&start %to &end %by &step;
    %&macroC.(k=&ind5., m=10000);
  %end;
%mend;
{\bf {\huge $\cdot$} Calculate $\delta$.}
\begin{verbatim}
%let blist=4, 6, 8;
%dalta();
%let blist=6, 9, 12;
%dalta(i=3,t=3);
%let blist=12, 24, 36;
%dalta(i=4,t=4);
data dataplot;
    t=3;
        do a=0 to 4 by 0.001;
```

```
            delta_BIB_b6=a**2* 6/(2* t);
            delta_ref_b6=a**2* 6*(t-1)/(4* t**2);
            delta_BIB_b12=a**2* 12/(2* t);
            delta_ref_b12=a**2* 12*(t-1)/(4* t**2);
            output;
        end;
run;
%graphsetting;
proc gplot data=dataplot;
    plot (delta_BIB_b6 delta_ref_b6 delta_BIB_b12 delta_ref_b12) *a
        /overlay legend=legend1 vaxis=axis;
run;
```

- **Density of p values** $d_\delta(x)$

```
data g_p;
    t=3; b=6;
    do x = 0.001 to 0.999 by 0.001;
        y=finv(1-x, t-1, b-t+1, 0);
        z0=pdf('f', y, t-1, b-t+1, 0);
        z1=pdf('f', y, t-1, b-t+1, 0.08);
        z2=pdf('f', y, t-1, b-t+1, 0.33);
        z3=pdf('f', y, t-1, b-t+1,1);
        z4=pdf('f', y, t-1, b-t+1, 4);
        g_p1=z1/z0;
        g_p2=z2/z0;
        g_p3=z3/z0;
        g_p4=z4/z0;
        label g_p1='delta=0.08';
        label g_p2='delta=0.33';
        label g_p3='delta=1';
        label g_p4='delta=4';
        output;
    end;
```

```
run;
%graphsetting;
proc gplot;
    plot (g_p1 g_p2 g_p3 g_p4 ) * x/overlay legend=legend1 vaxis=axis;
run;
```

- **Comparison between $d_\delta(x)$ and its upper bound.**

```
data upper;
    input nu1 nu2 delta @@;
    upper=exp(delta*nu2/(2*nu1));
    cards;
    1 3 0.25 ... 3 21 48
    ; *input nu1, nu2, and delta. delta could be calculated from
    the first function;
run;
*density function of d(x) and upper bound;
%dp(t=2,b=4,delta=0.25);
%dp(t=2,b=4,delta=1);
%dp(t=2,b=4,delta=4);
%dp(t=2,b=6,delta=0.375);
%dp(t=2,b=6,delta=1.5);
%dp(t=2,b=6,delta=6);
%dp(t=4,b=12,delta=0.375);
%dp(t=4,b=12,delta=1.5);
%dp(t=4,b=12,delta=6);
```

- **Calculate $d_\delta(x)$ with $F_0^{-1}(1-x) = 10000$ and 100000.**

```
*estimate the range of F^-1(0);
%macro est_F0(t=2, b=4, a=2, lambda=0.99);
data a;
    t=&t;   b=&b;   a=&a;   lambda=&lambda;
    delta_BIB =((a)**2) *b/(2* t);
    delta_Ref= ((a)**2)*b*(t-1)/(4*(t**2));
```

```
    do x=0.00001 to 0.00000101 by -0.000001;
        F_0=finv(1-x, t-1, b-t+1, 0);
        f0_BIB=pdf('f', F_0, t-1, b-t+1, 0);
        f1_BIB=pdf('f', F_0, t-1, b-t+1, delta_BIB);
        ratio_BIB=f1_BIB/f0_BIB;
        g_BIB=0.99+ratio_BIB*0.01;
        F_0_ref=finv(1-x, t-1, b-t, 0);
        f0_ref=pdf('f', F_0_ref, t-1, b-t, 0);
        f1_ref=pdf('f', F_0_ref, t-1, b-t, delta_ref);
        ratio_ref=f1_ref/f0_ref;
        g_ref=0.99+ratio_ref*0.01;
        output;
    end;
    do y=10000 to 100000  by 10000;
        f0_BIB=pdf('f', y, t-1, b-t+1, 0);
        f1_BIB=pdf('f', y, t-1, b-t+1, delta_BIB);
        ratio_BIB=f1_BIB/f0_BIB;
        g_BIB=lambda+(1-lambda)*ratio_BIB;
        f0_ref=pdf('f', y, t-1, b-t, 0);
        f1_ref=pdf('f', y, t-1, b-t, delta_ref);
        ratio_ref=f1_ref/f0_ref;
        g_ref=lambda+(1-lambda)*ratio_ref;
        output;
    end;
    *keep t b a delta x y F_0 ratio;
run;
proc print; run;
proc append base=outdata.d_delta_est data=a; run;
%mend;
%est_F0(t=2, b=4, a=1);
%est_F0(t=2, b=4, a=2);
%est_F0(t=2, b=6, a=1);
%est_F0(t=2, b=6, a=2);
```

```
...
%est_F0(t=10, b=180, a=1);
```

• **Simulations to estimate** $P[R = k]$.

```
dm 'log' clear;
* To simulate m p-values where lambda percent of them come from uniform
distribution and others from different alternative distributions;
* To run this program, few subdirectories need to be generated first;
* Each time to call the micro, REMEMBER to change the parameter ind=, otherwise
the previous data set might be overwritten;
* generate data;
%macro simu(ind1=1, design=BIB , t=2, b=3, delta=1, nu1=1, loop=1,
    q=0.1, m=10000, lambda=0.99);
%do j=1 %to &loop;
    data calculate_m0;
        bino=ranbin(0, &m, &lambda);
        call symput('m0', bino);
    run;
    %put &m0;
    *start simulation;
     %let m1=&m - &m0;
    %let k=0;
    data p_values1;
        length f $30;
        f="uniform";
        do i=1 to &m0;
            p=ranuni(0);                    *generate p-values under H0;
            output;
        end;
        drop i;
    run;
    data p_values2;
        f="F(&nu1,&nu2,&delta) with &design design";
```

```
    do l=1 to &m1;
        x=ranuni(0);
        z=finv(1-x, &nu1, &nu2, &delta);
        p=1-probf(z,&nu1,&nu2,0);       *generate p-values under Ha;
        output;
    end;
    drop l x;
run;
data p_values;
    set p_values1 p_values2;
run;
proc sort; by descending p; run;
data p_order; set p_values; by descending p;
    ind=&m - _n_ +1;
    cut=ind*&q/&m;
    diff=p-cut;
run;
data temp; set p_order;
    if diff <=0;
    keep ind diff;
run;
data cutoff; set temp;
    if _n_=1 then do;
         x=ind;
         call symput("k",x);  * k is the number of rejection;
         put x;
         end;
    else delete;
run;
%if &k > 0 %then %do;
    proc sort data=p_order; by p; run;
    data a; set p_order;
        if _n_ <= &k;
```

```
        run;
        proc freq data=a noprint; table f / out=outf; run;
        data outf; set outf; k=&k;  run;
        proc append base=outd&ind1..freqcy_&ind1
                data=outf;
        run;
        data K1; k=&k; run;
        proc append base=outd&ind1..Klist_&ind1
                data=K1;
        run;
    %end;
 %end;
 proc freq data=outd&ind1..Klist_&ind1 noprint;
    table k /out=outd&ind1..freqK ;
 run;
 data outd&ind1..freqK; set outd&ind1..freqK;
    prob=count/&loop;
    keep k count prob;
 run;
%mend;
%macro simulation(ind1=1, design=BIB , t=4, b=12, a=1, sigma2=1, loop=10000,
    q=0.1, m=10000, lambda=0.99);
* generate parameters for F test statistics for different designs;
 %let nu1=%eval(&t-1);
 %if &design=BIB   %then %do;
    %let nu2=%eval(&b-&t+1);
    %let delta=%sysevalf(((&a)**2) *&b/(2*&sigma2*&t));
    %end;
 %else %if &design=Ref   %then %do;
    %let nu2=%eval(&b-&t);
    %let delta=%sysevalf(((&a)**2)*&b*(&t-1)/(4*&sigma2*(&t**2)));
    %end;
 %simu(ind1=&ind1, design=&design, t=&t, b=&b, nu1=&nu1, delta=&delta, loop=&loop,
```

```
    q=&q, m=&m, lambda=&lambda);
%mend;
libname outd1 "C:\freq1";
%simulation(ind1=1, design=BIB , t=4, b=24, a=1, sigma2=1, loop=10000,
    q=0.1, m=10000, lambda=0.99);
*call the macro for different t, b, and a;
```

- **Simulation result.**

```
libname freq1 "C:\freq1";
data freq1; set freq1.freqk;
    design="BIB"; t=4; b=24; a=1; loop=10000; m=10000; lambda=0.99 ;
run;
...
libname freq523 "C:\freq523";
data outd523; set outd523.freqk;
    design="BIB"; t=10; b=270; a=0.5; loop=10000; m=10000; lambda=0.99 ;
run;
data Simulation_0_freq_k_BIB;
    set   freq1 ... outd523;
run;
** add prob0 and cumulative prob for bib design;
proc sort data=Simulation_0_freq_k_BIB; by t b a lambda loop m k ;run;
proc summary data=Simulation_0_freq_k_BIB;
    by t b a lambda loop m;
    var prob count;
    output out=prob_sum sum=prob_non0 count1;
run;
proc sort; by t b a lambda loop m; run;
data prob0; set prob_sum;
    k=0; design="BIB";
    count=loop-count1;
    prob=1-prob_non0;
    drop prob_non0 _type_ _freq_ count1;
```

```
run;
data Simulation_0_freq_k_BIB2; set Simulation_0_freq_k_BIB prob0; run;
proc sort;  by t b a lambda loop m k ; run;
data  outdata.Simulation_0_freq_k_BIB_v10 ;
    set Simulation_0_freq_k_BIB2;
    by t b a lambda loop m ;
    prob_cum+prob;
    if first.m then prob_cum=prob;
run;
```

- **Estimated the upper bound of $P[R = k]$.**

```
%macro est_k_L5_alt(k=1, q=0.1, m=10000, lambda=0.99);
data  Prob_&k;
    k=&k;
    cons=q/m;
    %text(s=0, e=5);
    A0=(x0*lambda +(1-lambda)* v0)**k;
    do r1=0 to 1;
        A1=(lambda*cons + (1-lambda)*(v2-v1))**r1;
        do r2=0 to 2-r1;
            A2=(lambda*cons + (1-lambda)*(v3-v2))**r2;
            do r3=0 to 3-r1-r2;
                A3=(lambda*cons + (1-lambda)*(v4-v3))**r3;
                do r4=0 to 4-r1-r2-r3;
                    A4=(lambda*cons + (1-lambda)*(v5-v4))**r4;
    l=&k+5;   *change for different L;
    sum_ri=r1+r2+r3+r4;
         A5=(lambda*(1-l*q/m)+ (1-lambda)*(1-v5))**(m-k-sum_ri);
    Prob_r_sub=((((A0*A1*A2*A3*A4*A5*comb(m, k+sum_ri)*gamma(k+sum_ri+1)
                /gamma(k))/gamma(r1+1))/gamma(r2+1))/gamma(r3+1))
                /gamma(r4+1);
    Prob_r+Prob_r_sub;
    end;    end;    end;    end;
```

```
run;
data Prob_&k; set Prob_&k;
     t=&t; b=&b; a=&a;
     keep t b a k m lambda q delta prob_r;
run;
proc append base=prob_k data=prob_&k; run;
%mend;
%prob_theory_L(q=0.1,t=2, b=4, a=0.5, macroC=est_k_L5_alt);
%prob_theory_L(q=0.1,t=2, b=4, a=1, macroC=est_k_L5_alt);
%prob_theory_L(q=0.1,t=2, b=4, a=2, start=1, end=30, macroC=est_k_L5_alt);
...
%prob_theory_L(q=0.1,t=10, b=270, a=2, start=1, end=30, macroC=est_k_L5_alt);
proc sort data=prob_K;
    by t b a  k ;
run;
%macro est_k_L10_alt(k=1, q=0.1, m=10000, lambda=0.99);
data  Prob_&k._L10;
    k=&k;
    cons=q/m;
    %text(s=0, e=10);
    A0=(x0*lambda +(1-lambda)* v0)**k;
    do r1=0 to 1;
        A1=(lambda*cons + (1-lambda)*(v2-v1))**r1;
        do r2=0 to 2-r1;
            A2=(lambda*cons + (1-lambda)*(v3-v2))**r2;
            do r3=0 to 3-r1-r2;
                A3=(lambda*cons + (1-lambda)*(v4-v3))**r3;
                do r4=0 to 4-r1-r2-r3;
                    A4=(lambda*cons + (1-lambda)*(v5-v4))**r4;
                    do r5=0 to 5-r1-r2-r3-r4;
                        A5=(lambda*cons + (1-lambda)*(v6-v5))**r5;
                        do r6=0 to 6-r1-r2-r3-r4-r5;
                            A6=(lambda*cons + (1-lambda)*(v7-v6))**r6;
```

```
                                do r7=0 to 7-r1-r2-r3-r4-r5-r6;
                                    A7=(lambda*cons + (1-lambda)*(v8-v7))**r7;
                                    do r8=0 to 8-r1-r2-r3-r4-r5-r6-r7;
                                        A8=(lambda*cons + (1-lambda)*(v9-v8))**r8;
                                        do r9=0 to 9-r1-r2-r3-r4-r5-r6-r7-r8;
                                            A9=(lambda*cons + (1-lambda)*(v10-v9))**r9;
    l=&k+10;   *change for different L;
    sum_ri=r1+r2+r3+r4+r5+r6+r7+r8+r9;
         A10=(lambda*(1-l*q/m)+ (1-lambda)*(1-v10))**(m-k-sum_ri);
    Prob_r_sub=(((((((((A0*A1*A2*A3*A4*A5*A6*A7*A8*A9*A10*comb(m, k+sum_ri)
                *gamma(k+sum_ri+1)/gamma(k))/gamma(r1+1))/gamma(r2+1))
                /gamma(r3+1))/gamma(r4+1))/gamma(r5+1))/gamma(r6+1))
                /gamma(r7+1))/gamma(r8+1))/gamma(r9+1);
    Prob_r+Prob_r_sub;
    *output;
    end;    end;    end;    end;
    end;    end;    end;    end;   end;
run;
data Prob_&k._L10; set Prob_&k._L10;
     t=&t; b=&b; a=&a;
     keep t b a k m lambda q delta prob_r;
run;
proc append base=prob_k_L10 data=prob_&k._L10; run;
%mend;
%prob_theory_L(q=0.1,t=2, b=4, a=2, start=1, end=10, macroC=est_k_L10_alt);
...
%prob_theory_L(q=0.1,t=10, b=270, a=2, start=100, end=110, macroC=est_k_L10_alt);
data simu; set outdata.Simulation_0_freq_k_BIB_v10;
proc sort data=simu;
    by t b a   k  ;
run;
proc sort data=prob_k_l10; by t b a k; run;
proc sort data=prob_k; by t b a k; run;
```

```
data BIB_table_l10;
    merge prob_k_l10(rename=(prob_r=upper_bound_R_equal_k_l10) in=in1)
          prob_k (rename=(prob_r=upper_bound_R_equal_k_l5) in=in1)
          simu (rename=prob=simu_R_equal_K);
     format upper_bound_R_equal_k_l5 upper_bound_R_equal_k_l10 simu_R_equal_K 7.6;
     by t b a  k ;
     if upper_bound_R_equal_k_l5 >1 then upper_bound_R_equal_k_l5=1;
     if upper_bound_R_equal_k_l10 >1 then upper_bound_R_equal_k_l10=1;
     if in1;
run;
proc sort; by t b a   k;   run;
proc transpose data=BIB_table_l10 out=BIB_tran_l10 prefix=k;
    by t b a    ;
    var simu_R_equal_K upper_bound_R_equal_k_l5 upper_bound_R_equal_k_l10;
    id k;
run;
proc export data=BIB_tran_l10
    outfile='C:\FDR\upper_bound_estimation_alter_R_equal_k.xls'
    dbms=excel replace
    ;
run;
```

• **Estimated the lower bound of $P[R > k]$.**

```
%macro est_k_L5_alt2(k=1, q=0.1, m=10000, lambda=0.99);
data  Prob_&k;
    k=&k;
    %text(s=1, e=5);
    cons=lambda*q/m;
    do r1=0 to &k;
        A1=(lambda*x1 + (1-lambda)*v1)**r1;
        do r2=0 to &k+1-r1;
            A2=(cons + (1-lambda)*(v2-v1))**r2;
            do r3=0 to &k+2-r1-r2;
```

```
                A3=(cons + (1-lambda)*(v3-v2))**r3;
                do r4=0 to &k+3-r1-r2-r3;
                    A4=(cons + (1-lambda)*(v4-v3))**r4;
                    do r5=0 to &k+4-r1-r2-r3-r4;
                        A5=(cons + (1-lambda)*(v5-v4))**r5;
    l=&k+5;   *change for different L;
    sum_ri=r1+r2+r3+r4+r5;
         A6=(lambda*(1-l*q/m)+ (1-lambda)*(1-v5))**(m-sum_ri);
    Prob_r_sub=((((A1*A2*A3*A4*A5*A6*comb(m, sum_ri)*gamma(sum_ri+1)
                /gamma(r1+1))/gamma(r2+1))/gamma(r3+1))/gamma(r4+1))
                /gamma(r5+1);
    Prob_r+Prob_r_sub;
    end;    end;    end;    end;    end;
run;
data Prob_&k; set Prob_&k;
     t=&t; b=&b; a=&a;
    right_side=1-prob_r;
run;
proc append base=prob_k data=prob_&k; run;
%mend;
%prob_theory_L(q=0.1,t=2, b=4, a=0.5, macroC=est_k_L5_alt2);
...
%prob_theory_L(q=0.1,t=10, b=270, a=2,start=80, end=110,step=10,
        macroC=est_k_L5_alt2);
proc sort data=prob_K;
    by t b a  k ;
run;
data outdata.lower_bound_R_lg_k_MN_L5;
    set prob_k;
run;

*estimate lower bound of P[R>k] by Poisson approximation;
%est_F0(t=2, b=4, a=0.5, lambda=0.99);
```

```
%est_F0(t=10, b=180, a=2, lambda=0.99);
%macro obtain_g;
    data _null_;
        set d_delta_est;
        if _n_ = &i then do;
            call symput('t', t);
            call symput('b',b);
            call symput('a', a);
            call symput('y', y);
                call symput('g1', g_BIB);
                call symput('delta', delta_BIB);
                call symput('d_delta', ratio_BIB);
                call symput('lambda', lambda);
        end;
    run;
%mend;
%macro est_k_L5_poisson(k=1, q=0.1);
data  Prob_&k._poisson;
    do r1=0 to &k;
        do r2=0 to &k+1-r1;
            do r3=0 to &k+2-r1-r2;
                do r4=0 to &k+3-r1-r2-r3;
                    do r5=0 to &k+4-r1-r2-r3-r4;
    g=&g; q=&q;
    l=&k+5;   *change for different L;
    sum_ri=r1+r2+r3+r4+r5;
    Prob_r_sub=(((((exp(-g*q*l)) *(g*q)**sum_ri *(&k+1)**r1/gamma(r1+1))
                /gamma(r2+1))/gamma(r3+1))/gamma(r4+1))/gamma(r5+1);
    Prob_r+Prob_r_sub;
    end;    end;    end;    end;    end;
run;
data Prob_&k._poisson; set Prob_&k._poisson;
    k=&k; design=&design; lambda=&lambda;
```

```
    t=&t; b=&b; a=&a;  g1=&g1;  delta=&delta;
    right_side=1-prob_r; d_delta_0_hat=&d_delta;
    keep g prob_r k right_side t b a  delta d_delta_0_hat design lambda g1 ;
run;
proc append base=prob_k_poisson data=prob_&k._poisson; run;
%mend;
%macro Lower_bound_L5_poisson(q=0.1, i1=1, i2=10, design='BIB', start=1,
            end=10, step=1);
%do i=&i1 %to &i2;
    %obtain_g;
    %let g=&g1;
    %do ind5=&start %to &end %by &step;
        %est_k_L5_poisson(k=&ind5. );
    %end;
%end;
%mend;
%Lower_bound_L5_poisson(q=0.1, i1=1, i2=45, design='BIB');
%Lower_bound_L5_poisson(q=0.1, i1=46, i2=46, design='BIB',start=15, end=55, step=5);
%Lower_bound_L5_poisson(q=0.1, i1=47, i2=47, design='BIB',start=15, end=35, step=5);
proc sort data=prob_k_poisson; by t b a k; run;
*obtain simulation result;
data simu; set outdata.Simulation_0_freq_k_BIB_v10;
proc sort data=simu;
    by t b a   k  ;
run;
data BIB_table;
    merge outdata.lower_bound_R_lg_k_MN_L5(rename=(right_side=lower_bound_R_L5_MN))
          prob_k_poisson(rename=(right_side=lower_bound_R_L5_PS))
          simu ;
     format prob_cum simu_R  lower_bound_R_L5_MN  lower_bound_R_L5_PS 6.4;
     by t b a  k ;
     simu_R=1-prob_cum;
run;
```

```
proc sort; by t b a   k;   run;
proc transpose data=BIB_table out=BIB_tran prefix=k;
    by t b a    ;
    var simu_R lower_bound_R_L5_MN  lower_bound_R_L5_PS;
    id k;
run;
proc export data=BIB_tran
    outfile='C:\FDR\lower_bound_R_lt_k.xls'
    dbms=excel replace
    ;
run;
```

• **Application of lower bound $P[R > k]$ for the BIB and the Ref designs.**

```
%macro calcu_v2(i=0);
    x&i.=(k+&i)*q/m;
    z&i.=finv(1-x&i.,&t-1, &b-&t, 0);
    v&i.=1-cdf('f', z&i., &t-1, &b-&t, delta);
%mend;
%macro est_k_L5_alt4(k=1, q=0.1, m=10000, lambda=0.99 );
data  Prob_&k;
    k=&k; m=&m; q=&q; lambda=&lambda;
    if &design='BIB ' then do;
        if &examp=1 then    delta =((&a)**2) *&b/(2* &t);
        if &examp=2 then delta=&b/(2*&t*(&t-1)) *(&t*(&a1**2+&a2**2+&a3**2) -
                                (&a1+&a2+&a3)**2);
        if &examp=3 then
            delta=&b/(2*&t*(&t-1)) *(&t*(&a1**2+&a2**2+&a3**2+&a4**2+&a5**2+
                    &a6**2+&a7**2)- (&a1+&a2+&a3+&a4+&a5+&a6+&a7)**2);
            %calcu_v(i=1);
            %calcu_v(i=2);
            %calcu_v(i=3);
            %calcu_v(i=4);
            %calcu_v(i=5);
```

```
    end;
if &design='Ref ' then do;
    if &examp=1 then    delta=((&a)**2)*&b*(&t-1)/(4*(&t**2));
    if &examp=2 then    delta=&b/(4* &t**2) *(&t*(&a1**2+&a2**2+&a3**2)
                                - (&a1+&a2+&a3)**2);
    if &examp=3 then    delta=&b/(4* &t**2) *(&t*(&a1**2+&a2**2+&a3**2+
        &a4**2+&a5**2+&a6**2 +&a7**2) - (&a1+&a2+&a3+&a4+&a5+&a6+&a7)**2);
        %calcu_v2(i=1);
        %calcu_v2(i=2);
        %calcu_v2(i=3);
        %calcu_v2(i=4);
        %calcu_v2(i=5);
    end;
if &design='Loop' then do;
    if &examp=2 then delta=&b/(2*&t) *(&a1**2+&a2**2+&a3**2 -(&a1*&a2+&a2*&a3) );
    if &examp=3 then delta=&b/(2*&t) *(&a1**2+&a2**2+&a3**2+&a4**2+&a5**2+&a6**2
                +&a7**2 - (&a1*&a2+&a2*&a3+&a3*&a4+&a4*&a5+&a5*&a6+&a6*&a7) );
        %calcu_v(i=1);
        %calcu_v(i=2);
        %calcu_v(i=3);
        %calcu_v(i=4);
        %calcu_v(i=5);
    end;
cons=lambda*q/m;
do r1=0 to &k;
    A1=(lambda*x1 + (1-lambda)*v1)**r1;
    do r2=0 to &k+1-r1;
        A2=(cons + (1-lambda)*(v2-v1))**r2;
        do r3=0 to &k+2-r1-r2;
            A3=(cons + (1-lambda)*(v3-v2))**r3;
            do r4=0 to &k+3-r1-r2-r3;
                A4=(cons + (1-lambda)*(v4-v3))**r4;
                do r5=0 to &k+4-r1-r2-r3-r4;
```

```
                    A5=(cons + (1-lambda)*(v5-v4))**r5;
    l=&k+5;   *change for different L;
    sum_ri=r1+r2+r3+r4+r5;
         A6=(lambda*(1-l*q/m)+ (1-lambda)*(1-v5))**(m-sum_ri);
    Prob_r_sub=((((A1*A2*A3*A4*A5*A6*comb(m, sum_ri)*gamma(sum_ri+1)
               /gamma(r1+1))/gamma(r2+1))/gamma(r3+1))/gamma(r4+1))
               /gamma(r5+1);
    Prob_r+Prob_r_sub;
    end;    end;    end;    end;    end;
run;
data Prob_&k; set Prob_&k;
     t=&t; b=&b; a=&a; design=&design; examp=&examp;
     a1=&a1; a2=&a2;a3=&a3;a4=&a4;a5=&a5;a6=&a6;a7=&a7;
    right_side=1-prob_r;
run;
proc append base=prob_k data=prob_&k; run;
%mend;
%macro lower_bound_R_L5(q=0.1,t=4, b=24, a=1, start=1, end=10, step=1 ,
    design='BIB ',examp=1, a1=1, a2=2, a3=3,a4=0, a5=0,a6=0,a7=0);
 %do ind5=&start %to &end %by &step;
    %est_k_L5_alt4(k=&ind5., m=10000);
  %end;
%mend;
*for example1, t=2, a=2;
%lower_bound_R_L5(q=0.1,t=2, b=4, a=2, design='BIB ' , examp=1);
...
%lower_bound_R_L5(q=0.1,t=2, b=30, a=2, design='Ref ', examp=1);
%macro outputRlt(out=t2a2 );
data prob_K; set prob_K;
    keep design m t b a examp a1 a2 a3 a4 a5 a6 a7 k lambda delta
        prob_r_sub prob_r right_side ;
    rename right_side=lower_bound_R_L5;
run;
```

```
proc sort data=prob_K;
    by design t b a examp a1 a2 a3 a4 a5 a6 a7 k;
run;
proc sort; by design t b a  examp a1 a2 a3 a4 a5 a6 a7 k;run;
proc transpose data=prob_K out=&out prefix=k;
    by design t b a examp a1 a2 a3 a4 a5 a6 a7   ;
    var   lower_bound_R_L5  ;
    id k;
run;
proc export data=&out
    outfile='C:FDR\lower_bound_R_lt_k_BIB_Ref.xls'
    dbms=excel replace
    ;
run;
%mend;
%outputRlt(out=t2a2 );
proc datasets; delete prob_k; run; *for example1, t=2, a=1;
%lower_bound_R_L5(q=0.1,t=2, b=4, a=1, design='BIB ', examp=1);
...
%lower_bound_R_L5(q=0.1,t=2, b=30, a=1, design='Ref ', examp=1);
*for example1, t=4, a=2;
%lower_bound_R_L5(q=0.1,t=4, b=6, a=2, design='BIB ', examp=1);
...
%lower_bound_R_L5(q=0.1,t=4, b=36, a=2, design='BIB ', examp=1);
*for example1, t=4, a=1;
%lower_bound_R_L5(q=0.1,t=4, b=6, a=1, design='BIB ', examp=1);
...
%lower_bound_R_L5(q=0.1,t=4, b=36, a=1, design='BIB ', examp=1);
*for example2, t=4 ;
%lower_bound_R_L5(q=0.1,t=4, b=12,  design='BIB ', examp=2, a1=1, a2=2,a3=3);
...
%lower_bound_R_L5(q=0.1,t=4, b=36, design='Loop', examp=2, a1=2, a2=1,a3=3);
%outputRlt(out=t4BIBLoop);
```

```
proc datasets; delete prob_k; run;
*for example3, t=8, b=56, 112 ;
%lower_bound_R_L5(q=0.1,t=8, b=56, design='BIB ', examp=3, a1=1, a2=3,a3=1,
    a4=3,a5=1,a6=3,a7=3, start=80, end=100, step=10);
%lower_bound_R_L5(q=0.1,t=8, b=56, design='Ref ', examp=3, a1=1, a2=3,a3=1,
    a4=3,a5=1,a6=3,a7=3, start=10, end=50, step=5);
%lower_bound_R_L5(q=0.1,t=8, b=56, design='Loop', examp=3, a1=1, a2=3,a3=1,
    a4=3,a5=1,a6=3,a7=3, start=10, end=50, step=5);
%outputRlt(out=t8LoopRef2);
data test;
    input a1 a2 a3;
    b=12; t=4;
     delta1= b/(2* t*( t-1)) *( t*( a1**2+ a2**2+ a3**2) - ( a1+ a2+ a3)**2);
     delta2= b/(4*  t**2) *( t*( a1**2+ a2**2+ a3**2) - ( a1+ a2+ a3)**2);
     delta3= b/(2* t) *( a1**2+ a2**2+ a3**2  - ( a1* a2+ a2* a3) );
    cards;
    1 2 1
    2 1 1
    0.5 2 0.5
    2 0.5 0.5
    1 2 0.5
    2 1 0.5
    ;
run;
proc datasets; delete prob_k; run;
*for example4, t=4 ;
%lower_bound_R_L5(q=0.1,t=4, b=12,  design='BIB ', examp=2, a1=1, a2=2,a3=1);
...
%lower_bound_R_L5(q=0.1,t=4, b=36, design='Loop', examp=2, a1=2, a2=1,a3=1,
            start=1, end=9, step=1);
%outputRlt(out=t4BIBLoop2);
```

• **Simulation for $P[S > j|m_1]$ is same as the simulation used to estimate $P[R = k]$ except that m_1 is fixed instead of having a binomial distribution.**

• **Estimated the lower bound of $P[S > j|m_1]$.**

```
%macro est_k_L5(j=0, q=0.1);
data  Prob_j_&j.;
    j=&j;
    n2=round(&m * (1-&lambda));
    cons=q/m;
    %text(s=1, e=5);
    do r1=0 to j;
        A1=v1**r1;
        do r2=0 to j+1-r1;
            A2=(v2-v1)**r2;
            do r3=0 to j+2-r1-r2;
                A3=(v3-v2)**r3;
                do r4=0 to j+3-r1-r2-r3;
                    A4=(v4-v3)**r4;
                    do r5=0 to j+4-r1-r2-r3-r4;
                            A5=(v5-v4)**r5;
    l=&j+5;   *change for different L;
    sum_ri=r1+r2+r3+r4+r5;
         A6=(1-v5)**(n2-sum_ri);
    Prob_j_sub=((((A1*A2*A3*A4*A5*A6*comb(n2,sum_ri)
            *gamma(sum_ri+1)/gamma(r1+1))/gamma(r2+1))/gamma(r3+1))
                /gamma(r4+1))/gamma(r5+1);
    Prob_j+Prob_j_sub;
    end;    end;    end;    end;    end;
run;
data Prob_j_&j; set Prob_j_&j;
     t=&t; b=&b; a=&a;
     keep t b a j m lambda q delta prob_j;
run; proc append base=prob_j data=prob_j_&j; run;
%mend;
%macro lower_bound_j_L5(q=0.1,t=4, b=24, a=1, start=0, end=10, step=1,
        m=10000, lambda=0.99);
```

```
    %do ind5 =&start %to &end %by &step;
        %est_k_L5(j=&ind5);
    %end;
%mend;
%lower_bound_j_L5(q=0.1,t=4, b=24, a=1);
...
%lower_bound_j_L5(q=0.1,t=8, b=168, a=2,start=70, end=90, step=10);

*get simulation results;
%macro getdata(t=4, b=24, a=1, loop=10000, ind=21, libn=freq, m=10000,
        lambda=0.99, design=BIB );
    %let nu1=%eval(&t-1);
    %let nu2=%eval(&b-&t+1);
    %let delta=%sysevalf(((&a)**2) *&b/(2*&t));
    data freqcy; set &libn.&ind..freqcy_&ind;
            j=count;
                if trim(left(f))="uniform" and percent=100 then
                    do;
                    f="F(&nu1, &nu2, &delta) with BIB design" ;
                    j=0;
                    output;
                    end;
                else if trim(left(f)) ne "uniform" then output;
            drop count percent;
    run;
    proc freq data=freqcy  noprint;
        table j / out=prob_S_simu1;
    run;
    proc summary data=prob_S_simu1;
        var count;
        output out=prob_S_0 sum=count1;
    run;
    data prob_S_0; set prob_S_0;
```

```
        j=0;  count_k0=&loop -count1; * for k=0;
        keep j   count_k0;
    run;
    proc sort data=prob_S_0; by j; run;
    proc sort data=prob_S_simu1; by j; run;
    data prob_S_simu2; merge prob_S_simu1 prob_S_0;
        by j;
        t=&t; b=&b; a=&a; m=&m; lambda=&lambda; design="&design";
        if j=0 then count=sum(count,count_k0);
        prob_simu_j=count/&loop;
        prob_simu_j_cum + prob_simu_j;
    run;
    proc append base=prob_S_simu data=prob_S_simu2;
    run;
%mend;
%macro assign_lib;
    %do ind=1 %to 30;
        libname fix&ind "C:\FDR\fix_m1_&ind.";
    %end;
%mend;
%assign_lib;
%getdata(t=4, b=24, a=1, loop=10000, ind=21);
...
%getdata(ind=50, design=BIB , t=8, b=168, a=2, loop=10000,
    m=10000, lambda=0.99,libn=fix);
*compair lower bound to simulation results;
proc sort data=Prob_j; by m lambda t b a  j; run;
proc sort data=prob_S_simu;
    by m lambda t b a   j  ;
run;
data BIB_j; merge prob_j prob_S_simu;
     format prob_j_lower_bound prob_simu_j  prob_simu_j_cum simu 7.4;
     by m lambda t b a  j;
```

```
     prob_j_lower_bound=1-prob_j;
     simu=1-prob_simu_j_cum;
run;
proc sort; by m lambda t b a j;   run;
proc transpose data=BIB_j out=BIB_tran_j prefix=j;
    by m lambda t b a    ;
    var   simu prob_j_lower_bound ;
    id j;
run;
proc export data=BIB_tran_j
    outfile="C:\FDR\Prob_S_given_n2_lower_bound.xls"
    dbms=excel replace
    ;
run;
```

• **Application of the lower bound of $P[S > j|m_1]$.**

```
*use L=5;
%lower_bound_j_L5(q=0.1,t=2, b=4, a=1);
...
%lower_bound_j_L5(q=0.1,t=2, b=20, a=2, start=60, end=90, step=10);
data prob_j; set prob_j   ;
     format prob_j_lower_bound  7.4;
     prob_j_lower_bound=1-prob_j;
run;
proc sort; by m lambda t b a j;   run;
proc transpose data=prob_j out=prob_tran_j prefix=j;
    by m lambda t b a    ;
    var prob_j_lower_bound ;
    id j;
run;
proc sort data=prob_tran_j; by  m lambda  a t b  ;   run;
proc export data=prob_tran_j
    outfile="C:\FDR\Prob_S_given_n2_lower_bound_example_for_sample_size.xls"
```

```
    dbms=excel replace
    ;
run;
```

• **Estimation of $P[\tilde{p}_{(k)} \leq x]$ and $P[\tilde{\tilde{p}}_{(k)} \leq x]$ for m original p-values, all under H_0^C or with $M_0 \sim bino(m, 0.99)$ under H_0^C while the rest $m - M_0$ under $H_a : K'T = a1$ with $a = 1, 2$ for different b. Probability are calculated from exponential approximation.**

```
%macro calc_adjustP_null;
    q1=betainv(x, 1,m);
    prob1=1-exp(-m*q1);
    q2=betainv(x, 1,m-1);
    prob2=prob1-m*q1*exp(-m*q2);
    q3=betainv(x, 1,m-2);
    prob3=prob2-m**2 *q1*exp(-m*q3)*(q2-q1/2);
    q4=betainv(x, 1,m-3);
    prob4=prob3-m**3 *q1*exp(-m*q4)*(q2*q3-q1*q3/2+q1**2/6-q2**2/2);
    output;
%mend;
%macro calc_adjustP_alter;
    q1=betainv(x, 1,m);
    prob1=1-exp(-m*q1*g);
    q2=betainv(x, 1,m-1);
    prob2=prob1-m*q1*exp(-m*q2*g)*g;
    q3=betainv(x, 1,m-2);
    prob3=prob2-m**2 *q1*exp(-m*q3*g)*(q2-q1/2)*(g**2);
    q4=betainv(x, 1,m-3);
    prob4=prob3-m**3 *q1*exp(-m*q4*g)*(q2*q3-q1*q3/2+q1**2/6-q2**2/2)*(g**3);
    output;
%mend;
*step-down Sidak; data sidak_null;
    m=10000;
    do x=0.01, 0.05, 0.1, 0.2;
```

```
        %calc_adjustP_null;
    end;
run;
proc print;
    var x prob1 prob2 prob3 prob4;
run;
%graphsetting;
data sidak2_null;
    m=10000;
    do x=0.0001 to 0.9999 by 0.0001;
        %calc_adjustP_null;
    end;
    keep x prob1-prob4;
run; proc gplot data=sidak2_null;
    plot (prob1 prob2 prob3 prob4)*x /overlay  vaxis =axis1;
run;
data modified_adjustP_null;
    m=10000;
    do x=0.01, 0.05, 0.1, 0.2;
        q1=betainv(x, 1,m);
        prob1=1-exp(-m*q1);
        q2=betainv(x, 2,m-1);
        prob2=prob1-m*q1*exp(-m*q2);
        q3=betainv(x, 3,m-2);
        prob3=prob2-m**2 *q1*exp(-m*q3)*(q2-q1/2);
        q4=betainv(x, 4,m-3);
        prob4=prob3-m**3 *q1*exp(-m*q4)*(q2*q3-q1*q3/2+q1**2/6-q2**2/2);
        output;
    end;
run;
proc print;
    var x prob1 prob2 prob3 prob4;
run;
```

```
%macro alter_sidak(t=2, a=1, b=4, m=10000, lambda=0.99);
*from exponential approximation;
data sidak_alter;
    m=&m; lambda=&lambda; t=&t; a=&a; b=&b;
    delta_BIB =((a)**2) *b/(2* t);
    y=10000 ;
        f0_BIB=pdf('f', y, t-1, b-t+1, 0);
        f1_BIB=pdf('f', y, t-1, b-t+1, delta_BIB);
        ratio_BIB=f1_BIB/f0_BIB;
        g=lambda +ratio_BIB*(1-lambda);
    do x=0.01, 0.05, 0.1, 0.2;
        %calc_adjustP_alter;
    end;
run;
proc print;
    var m lambda t b a delta_bib x prob1 prob2 prob3 prob4;
run;
%mend;
%alter_sidak(t=2,  b=4, a=1, m=10000, lambda=0.99);
...
%alter_sidak(t=2,  b=15, a=2, m=10000, lambda=0.99);

*Modify;
%macro alter_Modify(t=2, a=1, b=4, m=10000, lambda=0.99);
data Modify_alter;
    m=&m; lambda=&lambda; t=&t; a=&a; b=&b;
    delta_BIB =((a)**2) *b/(2* t);
    y=10000 ;
        f0_BIB=pdf('f', y, t-1, b-t+1, 0);
        f1_BIB=pdf('f', y, t-1, b-t+1, delta_BIB);
        ratio_BIB=f1_BIB/f0_BIB;
        g=lambda +ratio_BIB*(1-lambda);
    do x=0.01, 0.05, 0.1, 0.2;
```

```
        q1=betainv(x, 1,m);
        prob1=1-exp(-m*q1*g);
        q2=betainv(x, 2,m-1);
        prob2=prob1-m*q1*exp(-m*q2*g)*g;
        q3=betainv(x, 3,m-2);
        prob3=prob2-m**2 *q1*exp(-m*q3*g)*(q2-q1/2)*(g**2);
        q4=betainv(x, 4,m-3);
        prob4=prob3-m**3 *q1*exp(-m*q4*g)*(q2*q3-q1*q3/2+q1**2/6-q2**2/2)*(g**3);
        output;
    end;
run;
proc print;
    var m lambda t b a delta_bib x prob1 prob2 prob3 prob4;
run;
%mend;
%alter_Modify(t=2,  b=4, a=1, m=10000, lambda=0.99);
...
%alter_Modify(t=2,  b=15, a=2, m=10000, lambda=0.99);
```

• **Simulation results of $P[\tilde{p}_{(k)} \leq x]$ and $P[\tilde{\tilde{p}}_{(k)} \leq x]$ for m original p-values, all under H_0^C or with $M_0 \sim bino(m, 0.99)$ under H_0^C while the rest $m - M_0$ under $H_a : K'T = a1$ with $a = 1, 2$ for different b.**

```
*by simulation;
%macro simu(ind1=1, design=BIB , t=2, b=4, a=1, sigma2=1,  loop=1,  m=10000, lambda=0
%let nu1=%eval(&t-1);
%if &design=BIB   %then %do;
    %let nu2=%eval(&b-&t+1);
    %let delta=%sysevalf(((&a)**2) *&b/(2*&sigma2*&t));
    %end;
%do j=1 %to &loop;
    data calculate_m0;
        bino=ranbin(0, &m, &lambda);
        call symput('m0', bino);
```

```
    run;
    %let m1=&m - &m0;
    data p_values1;
        length f $30;
        f="uniform";
        do i=1 to &m0;
            p=ranuni(0);
            output;
        end;
        drop i;
    run;
    data p_values2;
        f="F(&nu1,&nu2,&delta) with &design design";
        do l=1 to &m1;
            x=ranuni(0);
            z=finv(1-x, &nu1, &nu2, &delta);
            p=1-probf(z,&nu1,&nu2,0);
            output;
        end;
        drop l x;
    run;
    data p_values;
        set p_values1 p_values2;
    run;
    proc sort; by  p; run;
    data p_values; set p_values;
        by  p;
        if _n_<=200;
        loop=&j;
    run;
    proc append base=WY.simulated_p_values_newset&ind1  data=p_values; run;
 %end;
%mend;
```

```
%simu(ind1=1,design=BIB , t=2, b=4, a=1, sigma2=1,  loop=10000,
    m=10000, lambda=0.99 );
...
%simu(ind1=8,design=BIB , t=2, b=15, a=2, sigma2=1,  loop=10000,
    m=10000, lambda=0.99 );
*calculate g;
%macro Calc_g(t=2, b=4, a=2);
data a;
    t=&t;    b=&b;    a=&a;
    delta_BIB =((a)**2) *b/(2* t);
     y=10000 ;
        f0_BIB=pdf('f', y, t-1, b-t+1, 0);
        f1_BIB=pdf('f', y, t-1, b-t+1, delta_BIB);
        ratio_BIB=f1_BIB/f0_BIB;
        g_BIB=0.99+ratio_BIB*0.01;
            output;
    keep t b a delta_BIB g_bib;
run;
proc append base=calc_g data=a; run;
%mend;
%Calc_g( t=2, b=4, a=1 );
...
%Calc_g( t=2, b=15, a=2 );
*calculate adjusted p-values;
%macro compare(alpha=0.05, var=p_adj, out=05);
data p_order2  ; set p_order1 ;
    if &var > &alpha then do;
        n_rej=n-1;
        output ;
    end;
    if n=200 and &var <= &alpha then do;
        n_rej=200;
```

```
        output;
    end;
run;
proc sort; by loop; run;
data WY.simulated_p_adj_rej_&ind1._&out;
    set p_order2;
    by loop;
    if first.loop;
    t=&t; b=&b; m=&m; lambda=&lambda; a=&a;
    keep t b m lambda a f p loop n n_rej &var;
run;
proc freq data=WY.simulated_p_adj_rej_&ind1._&out noprint;
    table n_rej/out=temp_&out;
run;
data temp_&out; set temp_&out;
    t=&t; b=&b; m=&m; lambda=&lambda; a=&a;
    pct_&out=percent;
    drop count percent;
run;
proc append base=WY.freq_newset_&out data=temp_&out; run;
%mend;
%macro adjustP( ind1=1, m=10000,  t=2, b=4, a=1, lambda=0.99 );
data p_order1 temp(keep=loop);
    set WY.simulated_p_values_newset&ind1;
    n=_n_-200*(loop-1);
    output p_order1;
    if p=. then output temp;
run;
proc sort data=p_order1; by loop p; run;
proc sort data=temp nodup; by loop; run;
data p_order1; merge p_order1 temp(in=in1);
    by loop  ;
    if in1 then delete;
```

```
    temp=cdf('beta', p, 1, &m-n+1);
    temp2=cdf('Beta', p, n,  &m-n+1);
run;
proc sort; by loop p; data p_order1; set p_order1;
    by loop p;
    retain I J 0;
    if first.loop  then do;
            I=temp; J=temp2;
    end;
    p_adj=max(I,temp); p_adj_m=max(j, temp2);
    I=p_adj; J=p_adj_m;
    drop i j temp temp2;
run; proc sort; by loop p; run;
%compare(alpha=0.01, var=p_adj, out=01);
%compare(alpha=0.05, var=p_adj, out=05);
%compare(alpha=0.1, var=p_adj, out=1);
%compare(alpha=0.2, var=p_adj, out=2);
%compare(alpha=0.01, var=p_adj_m, out=01m);
%compare(alpha=0.05, var=p_adj_m, out=05m);
%compare(alpha=0.1, var=p_adj_m, out=1m);
%compare(alpha=0.2, var=p_adj_m, out=2m);
%mend;
%adjustP(ind1=1, t=2, b=4, a=1, m=10000, lambda=0.99 );
...
%adjustP(ind1=8,  t=2, b=15, a=2, m=10000, lambda=0.99);
%macro merge1(ind1=01, datain1=WY.freq_newset_01, datain2=WY.freq_newset_01m,
            dataout=freq01);
proc sort data=WY.freq_newset_&ind1; by m lambda a t b n_rej; run;
proc sort data=WY.freq_newset_&ind1.m; by m lambda a t b n_rej; run;
data freq&ind1;
    merge WY.freq_newset_&ind1 WY.freq_newset_&ind1.m;
    by m lambda a t b n_rej;
run;
```

```
data  freq; set freq&ind1  ;
    by m lambda a t b ;
    if first.b then do; temp=pct_&ind1; temp_m=pct_&ind1.m; end;
    else do; temp+pct_&ind1; temp_m+pct_&ind1.m; end;
    pct_&ind1._cum=(100-temp)/100;
    pct_&ind1.m_cum=(100-temp_m)/100;
run;
proc transpose data=freq out=freq_pct_&ind1 prefix=k;
    by m lambda a t b ;
    var pct_&ind1._cum pct_&ind1.m_cum;
    id n_rej;
run;
proc export data=freq_pct_&ind1
    outfile="C:\WY\WY_table_beta.xls"
    dbms=excel replace;
run;
%mend;
%merge1(ind1=01);
%merge1(ind1=05);
%merge1(ind1=1);
%merge1(ind1=2);
```

• **Comparing** $P[Bino(m, p_{(k)}) \geq k]$ **and** $P[Bino(m-k+1, p_{(k)}) \geq 1]$.

```
data test;
    do m=1000, 10000;
        do i=2 to m;
        do p=0.00001 to 0.0001 by 0.00001;
            a=1-cdf('binomial', i-1, p, m);
            b=1-cdf('binomial',0, p, m-i+1);
            if a<=b then ind=1;
            output;
        end;
        do p=0.0001 to 0.001 by 0.0001;
```

```
            a=1-cdf('binomial', i-1, p, m);
            b=1-cdf('binomial',0, p, m-i+1);
            if a<=b then ind=1;
            output;
        end;
        do p=0.001 to 0.01 by 0.001;
            a=1-cdf('binomial', i-1, p, m);
            b=1-cdf('binomial',0, p, m-i+1);
            if a<=b then ind=1;
            output;
        end;
        do p=0.01 to 0.1 by 0.01;
            a=1-cdf('binomial', i-1, p, m);
            b=1-cdf('binomial',0, p, m-i+1);
            if a<=b then ind=1;
            output;
        end;
        do p=0.1 to 0.9 by 0.1;
            a=1-cdf('binomial', i-1, p, m);
            b=1-cdf('binomial',0, p, m-i+1);
            if a<=b then ind=1;
            output;
        end;
        end;
    end;
run;
proc freq ; table ind; run;
data test; set test;
    diff=b-a;
run;
proc sort; by p i m; run;
data test1; set test;
    where i in (1,2,3,4,5,6,7,8,9,10, 100, 200, 300,1000, 5000, 10000);
```

```
run;
proc sort data=test1; by m i p;
proc gplot; by m i;
    plot  diff*p;
run;
data test2; set test1;
    where p<0.01;
run;
proc sort data=test2; by m i p;
proc gplot data=test2(where=(m=10000 and p<0.001));
    by m i;
    plot  diff*p;
run;
```

• **Example to illustrate the difference between the step-down Sidak method and the modified adjusted p-values.**

```
%macro S_M_compare2(m=10, lambda=0.8, alpha=0.05, loop=10000);
%do j=1 %to &loop;
    data calculate_m0;
        bino=ranbin(0, &m, &lambda);
        call symput('m0', bino);
    run;
data tenGenes;
    do i=1 to &m0;
        p=ranuni(0);
        output;
    end;
    m1=&m-&m0;
    do j=1 to m1;
        x=ranuni(0) ;
        alt=1;
         z=finv(1-x, 2, 10, 10);
         p=1-probf(z,2, 10,0);
```

```
        output;
    end;
run;
proc sort data=tenGenes; by p; run;
data tenGenes; set tenGenes;
    m=&m;
    n=_n_;
    P_adj_s=cdf('beta', p, 1, m-n+1);
    P_adj_M=cdf('beta', p, n,  m-n+1);
    if P_adj_s < &alpha then rej_S=1;
    if P_adj_M < &alpha then rej_M=1;
run;
data temp1; set tenGenes;
    if rej_s=.;
    keep n rej_s;
run;
data _null_; set temp1;
    if _n_=1 then call symput("n_rej_s", n-1);
run;
data temp2; set tenGenes;
    if rej_M=.;
    keep n rej_M;
run;
data _null_; set temp2;
    if _n_=1 then call symput("n_rej_M", n-1);
run;
data temp;
    n_rej_s = &n_rej_s;
    n_rej_M = &n_rej_M;
    loop=&loop;
run;
proc append base=WY.nRej_delta data=temp; run;
%end;
```

```
%mend;
%S_M_compare2(m=10, lambda=0.8, alpha=0.05, loop=10000);
proc freq data=WY.nRej_delta;
    table n_rej_s*n_rej_M/nopercent;
run;
```

• **Analysis the HIV data.**

```
libname toy 'C:\toyData';
libname WY "C:\WY";

%macro readdata(datain=data1a, dataout=duplicate1);
PROC IMPORT OUT= &datain
            DATAFILE= "C:\toydata\&datain..xls"
            DBMS=EXCEL REPLACE;
     GETNAMES=YES;
RUN; *some genes are replicated more than twice on each slide;
proc sort data=&datain; by acc;
proc freq data=&datain noprint; table acc/out=temp2; run;
data &dataout; set temp2;
    if count > 2;
run;
%mend;
%readdata(datain=data1a, dataout=duplicate1);
%readdata(datain=data2a, dataout=duplicate2);
%readdata(datain=data3a, dataout=duplicate3);

*check duplicate in the three data sets;
proc sort data=duplicate1; by acc;run;
proc sort data=duplicate2; by acc;run;
proc sort data=duplicate3; by acc;run;
data duplicate;
    merge duplicate1 (rename=(count=count1) keep=count acc)
            duplicate2 (rename=(count=count2) keep=count acc)
```

```
            duplicate3 (rename=(count=count3) keep=count acc);
    by acc;
    ind1=count1-count2;
    ind2=count2-count3;
run;
proc freq; table ind1 ind2; run;
*duplicates are repeated in all three data sets in same way;

*for illustration, only use genes with 2 replicates per slide;
%macro geneSelect(datain1=duplicate1, datain2=data1a,
    dataout=data1a_1);
data dup1; set &datain1;  keep acc; run;
proc sort; by acc;
proc sort data=&datain2; by acc;
data &dataout; merge &datain2 dup1(in=in1);
    by acc;
    if in1 then delete;
run;
%mend;
%geneSelect(datain1=duplicate1, datain2=data1a, dataout=data1a_1);
%geneSelect(datain1=duplicate2, datain2=data2a, dataout=data2a_1);
%geneSelect(datain1=duplicate3, datain2=data3a, dataout=data3a_1);

*check the variations of the background intensities;
proc means data=data1a; var gbdata1a rbdata1a; run;
proc means data=data2a; var gbdata2a rbdata2a; run;
proc means data=data3a; var gbdata3a rbdata3a; run;
*large variations - use local background correction;

*************** background correction *************** ;
*check whether background need to be adjusted;
symbol1 line=1 c=black width=1.5;
symbol2 line=2 c=black width=1.5;
```

```
symbol3 line=3 c=black width=1.5;
symbol4 line=11 c=grey width=1.5;
symbol5 line=25 c=grey width=1.5;
symbol6 line=42 c=black width=1.5;
symbol7 line=34 c=black width=1.5;
symbol8 line=63 c=black width=1.5;
symbol9 line=17 c=black width=1.5;
%macro datamdfy(datain=data1a, gbvar=gbdata1a, gvar=gdata1a,
                rbvar=rbdata1a, rvar=rdata1a);
proc sort data=&datain; by x_location y_location; run;
data temp3; set &datain;
    by x_location y_location;
    if first.y_location ne last.y_location;
run;
data &datain; set &datain;
    by x_location y_location;
    lr=log2(&rvar);
    lrb=log2(&rbvar);
    lg=log2(&gvar);
    lgb=log2(&gbvar);
    lrdata=lr-lrb;
    lgdata=lg-lgb;
    if lrdata<0 then lrdata=0;
    if lgdata<0 then lgdata=0;
    M=lrdata -lgdata;
    A=lrdata +lgdata;
    if first.y_location = last.y_location;
run;
proc g3grid data=&datain out=data1;
   grid x_location*y_location= lr lrb lg lgb lrdata lgdata / join;
run;
proc gcontour  data=data1  ;
    title 'log transformed intensities';
```

```
    plot x_location*y_location= lg /autolabel ;
run;
proc gcontour  data=data1  ;
    title 'log transformed background ';
    plot x_location*y_location= lgb  ;
run;
proc gcontour  data=data1  ;
    title 'log intensities after background correction';
    plot x_location*y_location= lgdata ;
run;
proc gplot data=&datain;
    plot (lrdata  lgdata)*lrdata/overlay;
run;
proc gplot data=&datain;
    plot M*A;
run;
%mend;
%datamdfy(datain=data1a_1, gbvar=rbdata1a, gvar=rdata1a,
                rbvar=gbdata1a, rvar=gdata1a);
*slide 1 has colour flipped comparing to other two slides -
variables are renamed here;
%datamdfy(datain=data2a_1, gbvar=gbdata2a, gvar=gdata2a,
                rbvar=rbdata2a, rvar=rdata2a);
%datamdfy(datain=data3a_1, gbvar=gbdata3a, gvar=gdata3a,
                rbvar=rbdata3a, rvar=rdata3a);

data temp1; set data1a_1;
    lred1=lrdata; lgreen1=lgdata;
proc sort; by acc; data temp2; set data2a_1;
    lred2=lrdata; lgreen2=lgdata;
proc sort; by acc; data temp3; set data3a_1;
    lred3=lrdata; lgreen3=lgdata;
proc sort; by acc;
```

```
data temp; merge temp1 temp2 temp3; by acc;
title "without";
proc gplot; plot lred1*lred2;
proc gplot; plot lred1*lred3;
proc gplot; plot lgreen1*lgreen2;
proc gplot; plot lgreen1*lgreen3;run;


*************** colour correction *************** ;
%macro colour(datain=data1a_1, datamid=StatOut1, dataout=new1, smoothpar=0.1);
data &datain; set &datain; obs=_n_; run;
proc sort; by obs; run;
proc loess data=&datain;
      ods Output  OutputStatistics=&datamid;
      model M=A/degree=1 smooth=&smoothpar  residual;
   run;
proc gplot data=&datamid;
    *by smoothingparameter;
    plot  residual   *A/overlay ;
run;
proc sort; by obs; run;

data &dataout;
    merge &datamid
            &datain (keep=acc x_Location y_Location obs);
    by obs;
    lred=(residual+A)/2;
    lgreen=(-residual+A)/2;
run; proc gplot; plot (lgreen lred)*lred/overlay; run;
%mend;
%colour();
%colour(datain=data2a_1, datamid=StatOut2, dataout=new2);
%colour(datain=data3a_1, datamid=StatOut3, dataout=new3, smoothpar=0.2);
```

```
data temp1; set new1;
    lred1=lred; lgreen1=lgreen;
proc sort; by acc;
data temp2; set new2;
    lred2=lred; lgreen2=lgreen;
proc sort; by acc;
data temp3; set new3;
    lred3=lred; lgreen3=lgreen;
proc sort; by acc;
data temp; merge temp1 temp2 temp3; by acc;
title "";
proc gplot; plot lred1*lred2;
proc gplot; plot lred1*lred3;
proc gplot; plot lgreen1*lgreen2;
proc gplot; plot lgreen1*lgreen3;run;

**************** slide correction **************** ;
%macro slide(datain=new1, dataout=slide1);
data temp1 (rename=(lgreen=intensity) keep=lgreen acc )
     temp2 (rename=(lred=intensity) keep=lred acc )  ;
    set &datain;
run;
data &dataout; set temp1 (in=in1) temp2(in=in2);
    if in1 then T="green";
    if in2 then T=" red ";
run;
proc means data=&dataout n q1 mean median q3 noprint;
    var intensity;
    output out=s1 median=median q1=q1 q3=q3;
run;
data _null_; set s1;
    call symput("med", median);
```

```
    call symput("qrt1", q1);
    call symput("qrt3", q3);
run;
data &dataout; set &dataout;
    y=(intensity-&med )/ (&qrt3-&qrt1);
run;
proc means q1 mean median q3; var y; run;
%mend;
%slide();
%slide(datain=new2, dataout=slide2);
%slide(datain=new3, dataout=slide3);

********************** data analysis ******************;
 *generate p-values;
data toydata;
    set slide1 (in=in1) slide2(in=in2) slide3 (in=in3);
    if in1 then B=1;
    if in2 then B=2;
    if in3 then B=3;
run;
proc sort data=toydata; by acc T;run;
ods listing close;
proc glm data=toydata  ;
    by acc;
    class B T;
    model y =  B   T;
    contrast "Q" T 1 -1 ;
    ods output contrasts=P_value ;
run;
ods listing;

***************** using different multiple testing procedures;
proc sort data=P_value out=FDRdata;
```

```
    by descending probF;
run;
data FDRdata; set FDRdata; by descending probF;
    m=9132;
    q=0.1;
    ind=m-_n_+1;
    cut=ind*q/m;
    diff=probF-cut;
run;
data temp; set FDRdata;
    if diff <=0;
    keep ind diff;
run;
data _null_; set temp;
    if _n_=1 then call symput("k",ind);
run;
proc sort data=FDRdata; by probF; run;
data FDRdata; set FDRdata;
    if _n_ <= &k;
run;
*rej 14;

proc sort data=P_value out=WYdata;
    by  probF;
run;
data WYdata; set WYdata;
    m=9132; n=_n_;
    temp1=cdf('beta', probF, 1, m-n+1);
    temp2=cdf('beta', probF, n,  m-n+1);
run;
proc sort; by probF;
data WYdata; set WYdata;
    by probF;
```

```
    retain I J 0;
    if n=1 then do;
            I=temp1; J=temp2;
    end;
    p_adj=max(I,temp1); p_adj_m=max(j, temp2);
    I=p_adj; J=p_adj_m;
    drop i j ;
run;
```

- **Predict the sample size needed for the rat experiment.**

```
*assumed that 100 genes are significantly differentially
expressed;
libname app 'C:\toyData';
data a;
    sigma=1;  q=0.1; lambda=1-100/31042;
    do k=2 to 10;
        b=10*(k-1);
        delta=(10+2*k)*0.25/sigma;
        y=10000;
        z1=pdf('f', y, 5, b , 0);
        w1=pdf('f', y, 5, b , delta);
        ratio=w1/z1;
        g=lambda+(1-lambda)*ratio;
        output;
    end;
    drop y z1 w1 ratio;
run;
proc print; var k delta g; run;

*FDR method - estimate lower bound of P[R>k] using multinomial;
%macro calcu_v3(i=0);
    x&i.=(k+&i)*q/m;
```

```
    z&i.=finv(1-x&i.,&nu1, &nu2, 0);
    v&i.=1-cdf('f', z&i., &nu1, &nu2, delta);
%mend;
%macro est_k_L5_alt4(k=1, sigma=1 );
data  Prob_&k;
    k=&k; m=&m; q=&q; lambda=&lambda; lsize=&lsize;
    %let nu1=5;
    %let nu2=10*(&lsize-1);
    delta=(10+2*&lsize)*0.25/&sigma;

    %calcu_v3(i=1);
    %calcu_v3(i=2);
    %calcu_v3(i=3);
    %calcu_v3(i=4);
    %calcu_v3(i=5);

    cons=lambda*q/m;
    do r1=0 to &k;
        A1=(lambda*x1 + (1-lambda)*v1)**r1;
        do r2=0 to &k+1-r1;
            A2=(cons + (1-lambda)*(v2-v1))**r2;
            do r3=0 to &k+2-r1-r2;
                A3=(cons + (1-lambda)*(v3-v2))**r3;
                do r4=0 to &k+3-r1-r2-r3;
                    A4=(cons + (1-lambda)*(v4-v3))**r4;
                    do r5=0 to &k+4-r1-r2-r3-r4;
                        A5=(cons + (1-lambda)*(v5-v4))**r5;
    l=&k+5;   *change for different L;
    sum_ri=r1+r2+r3+r4+r5;
         A6=(lambda*(1-l*q/m)+ (1-lambda)*(1-v5))**(m-sum_ri);
    Prob_r_sub=((((A1*A2*A3*A4*A5*A6*comb(m, sum_ri)*gamma(sum_ri+1)
```

```
                /gamma(r1+1))/gamma(r2+1))/gamma(r3+1))/gamma(r4+1))
                /gamma(r5+1);
    Prob_r+Prob_r_sub;
    end;    end;    end;    end;    end;
run;
data Prob_&k; set Prob_&k;
    right_side=1-prob_r;
run;
proc append base=prob_k data=prob_&k; run;
%mend;
%macro lower_bound_R_L5_pred(q=0.1, start=1, end=5, step=1, lsize=2,
        m=31042, lambda=0.997);
 %do ind5=&start %to &end %by &step;
    %est_k_L5_alt4(k=&ind5.);
  %end;
%mend;

%lower_bound_R_L5_pred(q=0.1, start=0, end=5, step=1, lsize=2,
        m=31042, lambda=0.997);
...
%lower_bound_R_L5_pred(q=0.1, start=0, end=5, step=1, lsize=10,
    m=31042, lambda=0.997);
data prob_K; set prob_K;
    keep lsize k lambda delta prob_r_sub prob_r right_side ;
    rename right_side=lower_bound_R_L5;
run;
proc sort data=prob_K;
    by lsize k;
run;
proc transpose data=prob_K out=temp prefix=k;
    by lsize ;
```

```
    var   lower_bound_R_L5  ;
    id k;
run;

*Step-down Sidak;
%macro app(l=4, sigma=1,  loop=1,  m=31042, lambda=0.997 );
%let nu1=5;
%let nu2=10*(&l-1);
data _null_;
    delta=(10+2*&l)*0.25/&sigma;
    call symput('delta', delta);
run;
%do j=1 %to &loop;
    data calculate_m0;
        bino=ranbin(0, &m, &lambda);
        call symput('m0', bino);
    run;
    *start simulations;
    %let m1=&m - &m0;
    data p_values1;
        length f $30;
        f="uniform";
        do i=1 to &m0;
            p=ranuni(0);                    *generate p-values under H0;
            output;
        end;
        drop i;
    run;
    data p_values2;
        f="F(&nu1,&nu2,&delta)";
        do l=1 to &m1;
```

```
            x=ranuni(0);
            z=finv(1-x, &nu1, &nu2, &delta);
            p=1-probf(z,&nu1,&nu2,0);       *generate p-values under Ha;
            output;
        end;
        drop l x;
    run;
    data p_values;
        set p_values1 p_values2;
    run;
    proc sort; by  p; run;
    data p_values; set p_values;
        by  p;
        if _n_<=200;
        loop=&j;
    run;
    proc append base=app.simulated_p_values_N&l  data=p_values; run;
 %end;
%mend;
%app(l=4, sigma=1,  loop=10000,  m=31042, lambda=0.997 );
%app(l=5, sigma=1,  loop=10000,  m=31042, lambda=0.997 );
%app(l=10, sigma=1,  loop=10000,  m=31042, lambda=0.997 );
%macro compare2(alpha=0.05, var=p_adj, out=05);
data p_order2  ; set p_order1 ;
    if &var > &alpha then do;
        n_rej=n-1;
        output ;
    end;
    if n=200 and &var <= &alpha then do;
        n_rej=200;
        output;
```

```
    end;
run;
proc sort; by loop; run;
data simulated_p_adj_rej_&out; set p_order2;
    by loop;
    if first.loop;
    l=&l;
    keep l f p loop n n_rej &var;
run;
proc freq data=simulated_p_adj_rej_&out noprint;
    table n_rej/out=temp_&out;
run;
data temp_&out; set temp_&out;
    pct_&out=percent;
    drop count percent;
run;
proc append base=app.freq_newset_&out data=temp_&out; run;
%mend;
%macro adjustP2(m=31042, lambda=0.997, l=4 );
data p_order1 temp(keep=loop); set app.simulated_p_values_N&l;
    n=_n_-200*(loop-1);
    output p_order1;
    if p=. then output temp;
run;
proc sort data=p_order1; by loop p; run;
proc sort data=temp nodup; by loop; run;
data p_order1;
    merge p_order1 temp(in=in1);
    by loop  ;
    if in1 then delete;
    temp=cdf('beta', p, 1, &m-n+1);
```

```
run;
proc sort; by loop p; data p_order1; set p_order1;
    by loop p;
    retain I 0;
    if first.loop  then do;
            I=temp;
    end;
    p_adj=max(I,temp);
    I=p_adj;
    drop i temp;
run;
proc sort; by loop p; run;
%compare2(alpha=0.1, var=p_adj, out=1);
%mend;
%adjustP2(m=31042, lambda=0.997, l=4 );
%adjustP2(m=31042, lambda=0.997, l=5 );
%adjustP2(m=31042, lambda=0.997, l=10 );
```

Bibliography

[1] Allison,D.B., Gradbury,G.L., Heo,M., Fernanfex,J.R., Lee,C., Prolla,T.A., and Weindruch,R., (2002), "A mixture model approach for the analysis of microarray gene expression data", Computational Statistics and Data Analysis 39:1-20.

[2] Allison,D.B., Page,G.P., Beasley,T.M., Edwards,J.W., (2005), "DNA Microarrays and Related Genomics Techniques", Chapman and Halll.

[3] Benjamini,Y. and Hochberg,Y., (1995), "Controlling the false discovery rate - a practical and powerful approach to multiple testing", J.Roy.Stat.Soc. Series B, Vol 57, No.1, 289-300.

[4] Brazma,A., Hingamp,P., Quackenbush,J., Sherlock,G., Spellman,P., Stoeckert,C., Aach,J., Ansorge,W., et al. (2001), "Minimum information about a microarray experiemnt (MIAME) - toward standards for microarray data", Nat. Genet. 29:365 - 371.

[5] Brown,P.O., and Botstein,D., (1999), "Exploring the New World of the Genome with DNA Microarrays", Nat. Genet. 21: 33-37.

[6] Clote,P., (2000), "Computational molecular biology: an introduction", John Wiley.

[7] Cui,X., and Churchill,G.A, (2003), "Statistical Tests for Differential Expression in cDNA Microarray Experiments", Genome Biology 4:210.1-210.9.

[8] Dabrowski,A., (2006), "Statistics for Comparison of Two Independent cDNA Filter Microarrays", preprint.

[9] Dabrowski,A., Laneuville,Trudel, "Preliminary data on genetic expression - background", preprint.

[10] Draghici,S, (2003), "Data Analysis Tools for DNA Microarrays", Chapman & Hall.

[11] Dudoit,S., Yang,Y.H., Callow,M.J. and Speed,T.P., (2002), "Statistical Methods for Identifying Differentially Expressed Genes in Replicated cDNA Microarray Experiments", Statist. Sinica, 12, 111-139.

[12] Efron,B., (2005), "Local False Discovery Rates", www-stat.stanford.edu/~brad /papers.

[13] Efron,B., Tibshirani,R', Storey,J.D., Tusher,V., (2001), "Emperical Bayes analysis of a microarray experiment", JASA 2001, 96:1151-1160.

[14] Haggis,G.H., (1964), "Introduction to molecular biology", Longmans.

[15] Hastie,T., Tibshirani,R., Friedman,J., (2001), "The Elements of Statistical Learning: Data Mining, Inference, and Prediciton", Springer.

[16] Hung,H.M., O'Neill,R.T., Bauser,P., and Kohne,K., (1997), "The Behavior of the p-value When the Alternative Hypothesis is Ture", Biometrics 53:11-22.

[17] John,P.W.M., (1971), "Statistical Design and Analysis of Experiments", MacMillian.

[18] Johnson and Kotz, (1970), "Distribution in Statistics: Continuous univariate distribution", V2, P191.

[19] Kerr,M.K., Martin,M., and Churchill,G.A., (2000), "Analysis of Variance for Gene Expression Microarray Data", j Comput Biol, 7: 819 - 837.

[20] Kerr,M.K., Churchill,G.A., (2001a), "Bootstrapping cluster analysis: Assessing the reliability of conclusions from microarray experiments", Proc Natl Acad Sci, Vol 98, NO.16, 8961-8965.

[21] Kerr,M.K., and Churchill,G.A., (2001b), "Experimental design for gene expression microarrays", Biostatistics, 2.2: 183 - 201 .

[22] Kerr,M.K., and Churchill,G.A., (2001c), "Statistical Design and the Analysis of Gene Expression Microarray Data", Genet.Res.77: 123-128.

[23] Labbe,A., (2005), "Multiple testing using the posterior probability of half-space: application to gene expression data", PH.D thesis in university of Waterloo.

[24] Leadbetter,M.R., Lindgren,G., Rootzen,H., (1983), "Extremes and Related Properties of Random sequences and Processes", Springer.

[25] Lee,M.T., and Whitmore,G.A., (2002), "Power and Sample Size for DNA Microarray Studies", Statist. Med., 21: 3543 - 3570.

[26] Lehmann,E.L., Pomano,J.P., and Shaffer,J.P., (2005), "On Optimality of Stepdown and Stepup Multiple Test Procecedures", The Annales of Statistics, V33, No.3, 1084-1108.

[27] Pounds,S., and Morris,S.W., (2003), "Estimating the Occurrence of False Positive and False Negative in Microarray Studies by Approximating and Partitioning the Empirical Distribution of p-values", Bioinformatics 19: 1236-1242.

[28] Resnick.S.I., (1987), "Extreme Values, Regular Variation, and Point Process", Berlin Heidelberg New York: Springer.

[29] Searle,S.R., (1971), "Linear models", Wiley.

[30] Sebastiani,P., Gussoni,E., Kohane,I.S., Ramoni,M.R., (2003), "Statistical Challenges in Functional Genomics", Statistical Science, Vol 18, NO 1, 33-70.

[31] Westfall,P.H. and Young,S.S., (1993), "Resampling-Based Multiple Testing: Examples and Methods for p-Value Adjustment", Wiley, New York.

[32] Dorman,J., (2001), Lecture Notes for Course "Genetics in Epidemiology", University of Michigan.