NOTE TO USERS

DISSERTATION

MODELS AND METHODS FOR THE ANALYSIS OF MICROARRAY DATA: BEFORE AND AFTER THE FOLD CHANGE CALCULATION

Submitted by

Ann M. Hess

Department of Statistics

In partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2005

COLORADO STATE UNIVERSITY

April 5, 2005

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY ANN M. HESS ENTITLED MODELS AND METHODS FOR THE ANALYSIS OF MICROARRAY DATA: BEFORE AND AFTER THE FOLD CHANGE CALCULATION BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work

Phillip Chapman

Chun Man Lee

Laurie Stargell

Hariharan Iyer (Adviser)

Richard Davis (Department Head)

ABSTRACT

MODELS AND METHODS FOR THE ANALYSIS OF MICROARRAY DATA: BEFORE AND AFTER THE FOLD CHANGE CALCULATION

Microarrays allow scientists to monitor expression levels of thousands of genes simultaneously. Scientists use microarrays to find relative expression (fold change) under various conditions. While the use of microarrays is now widely accepted, there are many proposed methods for analyzing microarray data. In this dissertation, we develop a framework for comparing methods for microarray data analysis and determination of sample size. We address issues relating to microarray data preprocessing, estimability of quantities of interest and diagnostics for checking model assumptions. We also propose a systematic procedure for transcriptional regulation analysis.

We compare the performance of the most popular methods for analysis of oligo microarray data (Microarray Suite 5.0, RMA and dChip) using a simulation framework (SimArray). A simulation study is employed because it allows for the manipulation of many aspects of an experiment, including the number of arrays, amount and sources of variability, the proportion of genes that are affected under a certain experimental condition and the fold changes. We discuss SimArray's use as a sample size calculator which allows scientists to choose an appropriate number of microarrays for a given experiment based on power and false discovery rate. A unique feature of SimArray is that it begins with probe level information.

A number of data preprocessing steps are taken before a model is fit to microarray data. One such step is normalization, which attempts to correct for systematic

array differences. We closely examine some commonly used normalization methods and detail their limitations. We also propose a number of diagnostics that check the effectiveness of the preprocessing and the consistency between model assumptions and observed data.

We propose a unified model which would allow all preprocessing to be incorporated into a single model for the analysis of microarray data. We discuss this proposed unified model and its relationship to other models.

Finally, we present a case study involving transcriptional regulation analysis, which uses estimated fold changes as input.

Ann M. Hess
Department of Statistics
Colorado State University
Fort Collins, Colorado 80523
Spring 2005

ACKNOWLEDGEMENTS

Firstly, I would like to thank Dr. Hari Iyer for serving as my adviser, teacher and friend during the completion of this dissertation. When I think of all I have learned as a graduate student, it seems I learned the most important lessons from him. It was under his guidance that I taught my first class, gave my first conference presentation, worked as a research assistant and completed both my M.S. and Ph.D. research. He continues to inspire me with his enthusiasm, knowledge and perspective.

Special thanks to Dr. Laurie Stargell for teaching me the basics of molecular biology, microarrays and transcriptional regulation. She offered me instruction, encouragement and data at the beginning stages of this dissertation when I most needed the help.

I would like to thank Dr. Phil Chapman and Dr. Thomas Lee for serving on my committee and for offering valuable suggestions on this dissertation.

Without the use of microarray data this thesis would not have been possible. I would like to thank Dr. Laurie Stargell and Dr. Sue Kramer for the use of the TBP data. I thank Dr. Gail Willsky, Dr. Debbie Crans and Dr. L.H. Chi for the use of the Vanadyl Sulfate diabetes data. I also thank Dr. Nora Lapitan and Dr. Anna-Maria Oberholster for the use of the barley data.

DEDICATION

To Erich, with your help I was able to accomplish two things at once. I never would have made it this far without your support. Thank you.

CONTENTS

ABBREVIATIONS

A	adenine
C	cytosine
cDNA	complementary DNA
CWER	comparison wise error rate
D	decreased
DNA	deoxyribonucleic acid
FC	fold change
FDR	false discovery rate
FWER	family wise error rate
G	guanine
GSB	gene specific binding
I	increased
IM	ideal mismatch
IVSN	invariant set normalization
MAS	Microarray Suite
MBEI	model based expression index
MIPS	Munich Information Center for Protein Sequences
MM	mismatch
NC	no change
NE	not evaluable
mRNA	messenger RNA
NSB	nonspecific binding
OVR	over-representation ratio
PCR	polymerase chain reaction
PM	perfect match
RMA	robust multi-array average
RNA	ribonucleic acid
ROC	receiver operating characteristic
rRNA	ribosomal RNA
RSA	regulatory sequence analysis
T	thymine
TBP	TATA binding protein
tRNA	transfer RNA
TSR	total signal ratio
U	uracil

Chapter 1

INTRODUCTION

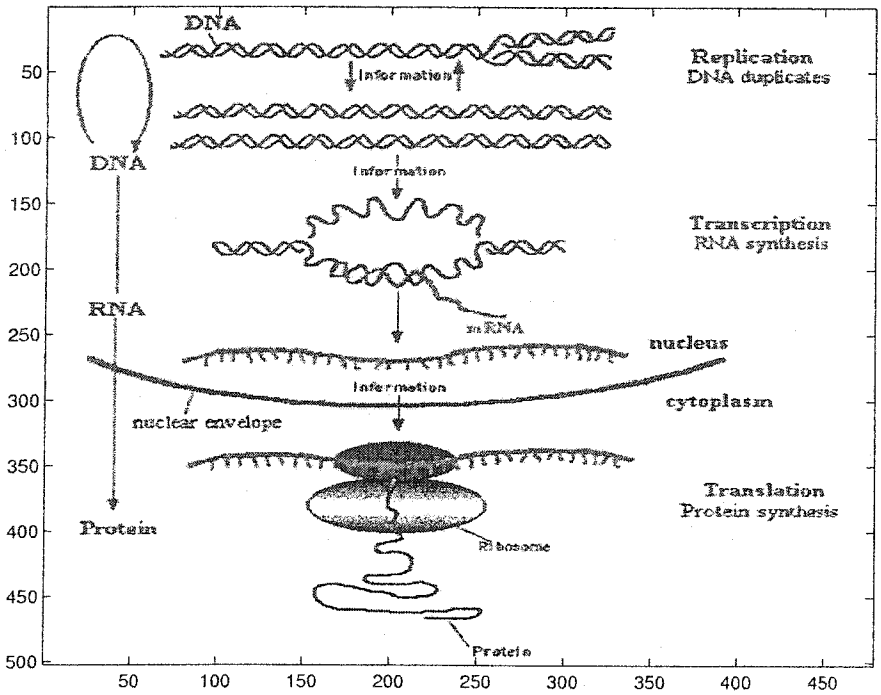
Microarrays allow scientists to monitor expression levels of thousands of genes simultaneously. In addition to their established use in research, microarrays are quickly becoming standard for diagnostic purposes. Here a brief introduction to molecular biology, bioinformatics and microarrays is provided, and some uses of microarrays are discussed.

In order to discuss microarray technology, we need to understand some basic molecular biology. The central dogma of molecular biology tells us that three steps are involved in storing and expressing genetic information. Genetic information is stored as DNA, but is expressed by the production of proteins. Replication is the process by which double stranded DNA is reproduced to form two identical copies. Transcription is the process by which information stored in DNA is copied to RNA. Finally, during translation a sequence of mRNA is used to generate a protein. A schematic of the central dogma is shown in Figure 1.1.

Deoxyribonucleic acid (DNA) is the basic genetic material. DNA is a double helix built with nucleic acids of bases (nucleotides) adenine (A), cytosine (C), guanine (G), and thymine (T). A pairs only with T and C pairs only with G. During replication, a strand of DNA breaks apart and two identical strands are created. DNA remains in the nucleus of the cell.

Different types of ribonucleic acid (RNA) produced during transcription include messenger RNA (mRNA), transfer RNA (tRNA) and ribosomal RNA (rRNA). mRNA contains the same information as DNA from which it was created, however

Figure 1.1: The Central Dogma of Molecular Biology



the nucleotide thymine (T) is replaced with uracil (U) and non-coding regions (those not used to construct a protein) are removed. mRNA travels from inside the nucleus to the cytoplasm where it can be translated into a protein. tRNA and rRNA are also involved in the construction of proteins. It is interesting to note that although mRNA contains the core genetic information, it makes up only a small percentage of total RNA. In the coding regions nucleotide triplets (codons) each represent a single amino acid, but an amino acid may be represented by more than one codon. Proteins are constructed from a sequence of amino acids.

In 1975, Ed Southern devised a method of detecting DNA fragments complementary to some RNA after the DNA had been separated by gel electrophoresis [64]. This process (called Southern blotting) provides expression level information for a few genes at a time. A similar procedure for RNA is termed northern blotting and the analogous procedure for proteins is called western blotting.

In 1995, Pat Brown and colleagues published a paper describing the first microarray [61]. The microarrays were created by spotting cDNA (complementary DNA) onto a glass slide. A two-color hybridization scheme was employed such that RNA from different sources were printed onto the same slide, but were labelled with different colored fluorescent dyes. Their results were verified by a Southern blot.

A number of advancements have allowed for development and improvement of microarrays. The use of glass slides (instead of a gel) allows for miniaturization and fluorescence based detection [38]. The rapidly growing database of DNA sequences allows scientists to explore a variety of genes across species. Another major contribution to the field was development of methods to construct many different oligonucleotides on a small slide [38]. An oligonucleotide (oligo) is a small piece of DNA. Unlike the cDNA arrays (developed by Pat Brown's group), oligo arrays employ multiple oligos from the same gene. In 1996, Affymetrix began sales of the GeneChip system which uses oligo arrays [1]. A picture of an Affymetrix GeneChip array is shown in Figure 1.2.

Figure 1.2: Affymetrix GeneChip array in hand.

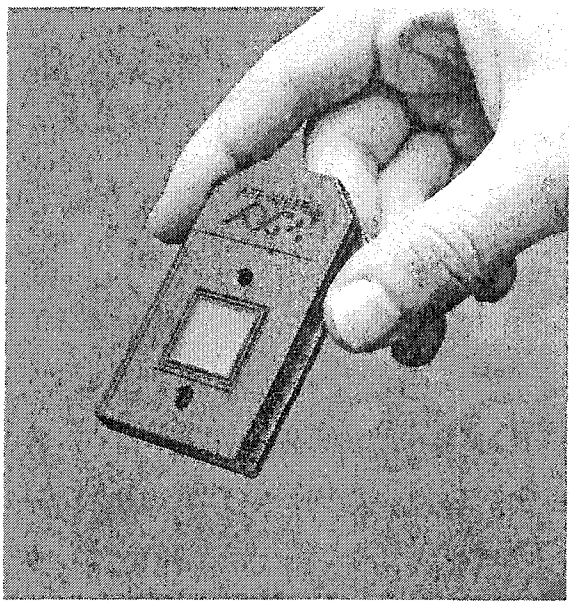


Image courtesy of Affymetrix.

The uses of microarray technology are diverse. Often the most fundamental question is whether or not a gene is differentially expressed under a certain condition as compared to some baseline. By clustering or grouping genes based on their expression levels across different conditions, scientists may gain information about gene function. Microarrays are also used to identify new disease classes (class discovery) and to assign subjects to known classes (class prediction) [25]. Oligo arrays are also being used to study DNA variants. The processes and mechanisms involved in transcription (transcriptional regulation) can be studied by comparing expression profiles of normal and mutated DNA with microarray technology.

It is important to remember that microarrays attempt to measure the amount of mRNA present. However, this provides only an indirect measure of protein production since an increase in the amount of a certain mRNA does not necessarily imply an increase in the corresponding protein. Alberts *et al.* discuss different methods of controlling protein production: "(1) controlling how or when a given gene is transcribed (transcriptional control), (2) controlling how the initial RNA transcript is processed (processing control), (3) selecting which completed mRNAs in the cell nucleus are exported to the cytoplasm (transport control), (4) selecting which mRNAs in the cytoplasm are translated by ribosomes (translational control), or (5) selectively stabilizing certain mRNA molecules in the cytoplasm (mRNA degradation control)" [4].

The analysis of microarrays is only a subset of the growing field of bioinformatics. Bioinformatics has been described as "the science of understanding the structure and function of genes and proteins through advanced, computer-aided statistical analysis and pattern discovery" [21]. Some of the original goals of bioinformatics were mapping of the genome and algorithms to compare a sequence of DNA (or protein) to a collected database. According to one author, the majority of recent effort has been directed towards protein identification (proteomics),

structure-function characterization (structural bioinformatics) and bioinformatics database mining [63].

Probably the most noted success of bioinformatics to date is the mapping of the human genome (preliminary sequencing completed in June 2000) [63]. However, mapping the genome is not the same as understanding it! There is still a wide gap between gene identification and gene function. One set of tools that is used in this pursuit is sequence alignment algorithms. Programs such as Smith-Waterman, BLAST and FASTA are used to compare DNA or protein sequences to known sequences, and identify recurring patterns (motifs).

Considerable focus is now being placed on the understanding of protein function. The study of protein function falls into the field of proteomics. Recall that although genetic information is stored in DNA it is expressed through the production of proteins. So called "rational drug design" is aimed at understanding protein design and function, so that improved proteins with specific therapeutic effects may be developed [63]. Similarly, structural bioinformatics concerns the physical structure of different proteins. Protein folding is the process by which a protein assumes its functional shape. Disruption of the functional shapes of proteins has been linked to such diseases as alzheimer's, cystic fibrosis and mad cow disease [68].

Data mining is used to harvest information from the large amount of data provided by experiments. One goal of data mining is to combine information from different microarray experiments (and other sources of information) to gain broader understanding of gene function. Another application of data mining is the identification of intron (coding regions) and exon (non-coding regions) boundaries in genomic DNA [63].

Statistical issues arise at many steps in the microarray process. Clearly statistics should be employed when designing microarray experiments. According to Lander, "the challenge is no longer in the expression arrays themselves, but in developing experimental designs to exploit the full power of a global perspective" [38].

Statistical image analysis is used to quantify the brightness of each "spot" on a microarray as an intensity value (roughly representing the amount of mRNA hybridized). After some further preprocessing of intensity values, a model is fit to the preprocessed values. "Fold change" calculations are carried out in an attempt to make comparisons within or between slides. Fold change estimates are used as the input for clustering or classification algorithms to gain perspective on the data and answer scientific questions.

The details of microarray experiments and sources of variation in these experiments are discussed in Chapter 2. Current methods for summarizing the intensity data into a gene expression index are discussed in Chapter 3. A comparison of methods (presented in Chapter 3) based on a detailed simulation experiment (SimArray) is given in Chapter 4. The use of SimArray as a sample size calculator is discussed in Chapter 5. Normalization methods are discussed in Chapter 6. In Chapter 7, we propose a unified model for oligonucleotide arrays. In Chapter 8, some probe level diagnostics and a test for differential expression are presented.

A discussion of the use of microarrays applied to transcriptional regulation analysis is discussed in Chapter 9 and a case study is provided in Chapter 10. Finally, conclusions and discussion of future work are given in Chapter 11.

Chapter 2

VARIATION IN MICROARRAYS

In this chapter we present a detailed discussion of microarray experiments. Our view starts even before the experiment, with a discussion of the life cycle of mRNA. It is the expression (or relative abundance) of this mRNA that is measured in a microarray experiment. The difference between oligonucleotide arrays and spotted cDNA arrays will be discussed. Sources of variation during the microarray process will be highlighted.

2.1 Life Cycle of Eukaryotic mRNA

Eukaryotic cells have a nucleus, while prokaryotic cells do not. There are major differences between the life cycles of prokaryotic and eukaryotic mRNA. Here we consider only eukaryotic mRNA.

For eukaryotic mRNA, transcription takes place in the nucleus. The rate of transcription is approximately 40 nucleotides per second [40]. A gene of length 10,000 base pairs takes about 5 minutes to transcribe, but multiple copies of a gene can be made at the same time. The RNA is not yet ready to leave the nucleus. First splicing occurs (when necessary) to remove non-coding regions (introns) yielding shorter mRNA with an "intact coding sequence". Lewin *et al.* [40] note that "producing an mRNA from an interrupted gene is the most labor-intensive of all RNA processing". It takes about 20 minutes for mRNA to leave the nucleus. Finally, the mRNA exits the nucleus to the cytoplasm.

Once in the cytoplasm, ribosomes begin to translate the mRNA. Eukaryotic mRNA makes up only about 3% (by mass) of total cellular RNA [40]. Half lives of mRNAs in animal cells are in the range of 4 to 24 hours. This allows time for the same strand of mRNA to be translated many times. Almost any ribosome can translate almost any mRNA. The ribosome at work on a given mRNA will usually restart translation on the same mRNA. There will be many ribosomes at work on the same mRNA.

However, mRNA is constantly degraded. According to Lewin *et al.* [40], "every mRNA is in statistical jeopardy at all times, with a constant probability that its decay will begin. Thus "young" mRNAs are as likely to be attacked as "old" mRNAs. Some copies of an mRNA are translated many times, while others function hardly at all. This random life expectancy is a feature of both prokaryotes and eukaryotes. But the overall translational yield of any messenger sequence is predictable."

2.2 Microarray Platforms

Microarrays are used to measure relative gene expression (based on mRNA abundance). Genes of interest are represented on the array using probes. mRNA is isolated from a given source, then reverse transcribed to cDNA. This cDNA (called the target) is dyed (with a fluorescent dye) and then hybridized onto the array.

Oligo Arrays: An oligonucleotide (oligo) is a short sequence of DNA. The oligos that are used for microarrays are usually 25 nucleotides long. Each gene of interest is represented by a number of different probe pairs. Such a group probe pairs is called a probe set. A probe pair consists of a PM (perfect match) and MM (mismatch) oligos. The PM probe is the exact complement to a section of DNA from the gene of interest. The MM probe is identical to the PM probe except at the central location (the 13th position for a 25 nucleotide probe). After cDNA is hybridized to the array, the MM probes are often used to account for nonspecific binding (meaning binding

to something other than the target transcript). Note that a probe set generally represents a gene, but not always. Oligo arrays are most often manufactured by companies (such as Affymetrix and Agilent) and therefore are more expensive to use than spotted cDNA arrays.

cDNA Arrays: For spotted cDNA arrays, longer probe sequences are used. cDNA from a target and reference sample are mixed in equal proportions and hybridized to the same array. The two samples are dyed using different fluorescent dyes (usually cy3 and cy5). After hybridization the array is scanned at two channels and intensities are recorded for both samples from a single array. The relative intensities appear as red and green spots. Since two samples appear on the same array, each array is self-normalizing. cDNA arrays are often prepared by individual labs, making them less expensive than oligo arrays.

Regardless of the type of array employed (oligo or cDNA), the goals of the analysis are usually the same. One study compares matched measurements from the two types of microarray technologies [37]. They found poor correlation between the technologies and differences in the clusters obtained. However, due to the fact that experiments in their study were carried out in multiple different labs, it is impossible to tell how much variation is from the technology and how much is from different lab protocols and individual technicians.

In a more recent study, Irizarry *et al.* [34] compare three different microarray platforms (Affymetrix GeneChips, two-color spotted cDNA arrays and two-color long oligo arrays). They first considered the accuracy of results from technical replicates at each of ten labs. It appears from their results that the lab effect was more important than the platform when assessing the accuracy of results based on technical replicates. To examine agreement between labs and platforms, they considered the proportion of probe sets that were jointly captured in the top X most differentially expressed genes (X=25,50 and 100) for two labs. They call this comparison

proportion of agreement. Using this criterion, they found that the Affymetrix platform was the most consistent across labs, with proportion of agreement greater than 50%. They also found good agreement across platforms for the more accurate labs.

2.3 Microarray Construction

For oligo arrays, individual researchers have relatively little control over the construction of the microarrays used in their experiment. Companies such as Affymetrix mass produce chips (arrays) for a number of different plants and animals and researchers simply purchase and use these chips. Custom arrays can be requested, but the construction is performed by the manufacturer. Affymetrix "uses masks to control synthesis of oligonucleotides on the surface of a chip" [36]. The oligo probes are constructed using a layered print-like process at specified locations on the chip.

For cDNA arrays, the construction of the microarray is more likely to be in the hands of the experimenter. The experimenter controls what probes are placed on the array, the isolation of the DNA sequences of interest and the placement of the spots onto the array. We now consider some issues involved in the construction of cDNA arrays.

The DNA sequences contained in the probes are often gene segments obtained by polymerase chain reaction (PCR) amplification. However, in addition to the DNA of interest, amplified samples also contain salts, enzymes, small DNA fragments, and other components [60, p203]. These contaminants can interfere with microarray experiments by clogging the pins and ink jets used to spot the DNA onto the slide; they may attach to the slide and interfere with hybridization (both at a specific spot or background hybridization) [60, p203]. PCR purification kits are available, and their use is recommended prior to microarray printing. Dye terminator clean-up kits are also recommended to accurately determine what DNA sequence is being spotted [60, p204].

Obtaining uniform spots on microarrays is very important. Wang *et al.* have found that variability in intensity ratio measurements is closely correlated with spot quality [72]. Spot uniformity means that an equal amount of DNA is present across the spot so that the binding rate will be equal across the spot and that there will be uniform signal intensities at each pixel [60, p204]. Uniformity of spots will be strongly influenced by the printing method used. Since such small amounts of DNA (in the nanoliter (10^{-9} L) or picoliter (10^{-12} L) range) are "spotted" at each location, printing technology has been developed specifically for microarrays. Rose discusses and compares different printing technologies in detail [60, p19]. He also points out that preprinting may be necessary to remove excess solution from the pin tip. He shows that after 10 to 20 preprint spots, the spotting becomes more consistent. This indicates that the order in which the spots are printed may also affect final results.

2.4 Sample Preparation

In order to obtain a sample of mRNA from a given source, cells from that source are harvested under given conditions. Next a phenol extraction is performed such that nucleic acids remain in the solution. A poly(T) column is performed so that mature mRNA (which have a poly(A) tail where the A nucleotide is repeated 50-200 times) are captured. Of course, a small amount of other items may still be present. According to Lewin *et al.* [40], "The poly(A) sequence is not coded in the DNA but is added to the RNA in the nucleus after transcription". This means that the presence or length of the poly(A) tail is independent of the gene from which the mRNA is produced. In addition a single enzyme (PolyA polymerase) is responsible for putting on the tails. A small percentage of mRNAs don't have a poly(A) tail and these are organism specific.

We would like equal amounts of mRNA from two (or more) samples. When harvesting cells under different conditions, it is difficult to be sure that the same

amount of mRNA is obtained. Generally, the same volume of cells yields about the same amount of mRNA. However, in order to check the concentrations ultraviolet lights can be employed (since nucleic acids absorb these rays). The use of the ultraviolet light quantifies the density of a solution and this value is then used to calculate the concentration using standards of known concentrations. Note that RNA is isolated from a tissue source and then the isolated RNA is converted to cDNA.

It is important to consider how mRNA abundance might vary at different stages of the life cycle of the cell. Information is available at different stages of this life cycle. It is generally assumed that total mRNA abundance is roughly fixed, while relative abundance of mRNA from different genes varies.

Schena and Davis note that for a single experiment it is essential for the samples to be processed in the same way [60, p5]. They also say that different commercial RNA isolation kits have yielded two fold or greater differences for as many as 1% of the human transcripts analyzed by microarray. In addition to the stated sample preparation method, the technician who prepares the sample will also influence the final results.

Another important step in the sample preparation is labelling of the samples. There are many methods available but they fall into two main categories: direct and indirect labelling [60, p7]. Direct labelling incorporates fluorescent dye directly into the sample that is hybridized to the array. With indirect labelling the fluorescent labelling occurs after hybridization.

Churchill and Kerr report that a dye by gene interaction is possible [46]. In one cDNA experiment, they found that spots for a single sequence (gene) on two different arrays had higher intensity on the green channel despite the fact that they had reversed the labelling of identical cDNA samples.

2.5 Microarray Reaction

Samples are hybridized to an array for 2 to 12 hours [60, p131]. Some important requirements for hybridization are: constant temperature, rapid temperature equilibrium, 100 percent humidity, pristine environment, low volumes of hybridization buffers and DNA [60, p211].

After hybridization has occurred, any excess (unbound) material is washed away. Even the rinsing process will play an important role in the detection step of the microarray experiment.

Clearly even when comparing replicate arrays there will be differences in the amount of binding that has occurred for a certain probe. Firstly, the available transcript abundances for each array may not be the same as the original sample. In addition, there may be other differences between arrays including labelling and binding efficiency differences.

2.6 Detection

For a cDNA array, the intensity levels for the two dyes (representing the two samples) are measured for each spot on the array. For an oligo array, only a single reading is obtained for each spot. In either case, these fluorescence values correspond to the level of hybridization to the DNA that has been spotted on the slide. Fluorescence detection and quantification are extremely important steps in a microarray experiment. All further results will be based on the values obtained at this step.

Scanners and charge-coupled devices are the two types of mechanisms used to image the slides. For each pixel, the digitization process produces an intensity value indicating the amount of fluorescence at that pixel. This intensity value should correspond to the density of dyed molecules in that region [78]. For cDNA arrays, this process is repeated at two channels, producing two images for each slide.

We would like our image to reflect the measure of the fluorescent intensities for the dye of interest and nothing else. However, the images contain undesired noise (photon noise, electronic noise, laser light reflection and background fluorescence) as well as the desired signal [78].

Image analysis begins with the digital images as the raw data. From this, we want to obtain an intensity measurement for each sample at each spot. First the spot locations are identified by gridding. Then segmentation classifies each pixel as foreground (belonging to the spot of interest) or background (not a specific spot). Intensity extraction includes calculating the foreground intensity measurements, background intensities and possibly quality measures for each spot on the array.

Many different algorithms and software programs are available for image processing, especially for cDNA arrays. Yang *et al.* compare different image analysis software programs including Spot, GenePix, ScanAlyze and QuantArray [78]. They find that the choice of background correction method has a larger impact than the segmentation method.

2.7 The Relationship between Intensity and Transcript Abundance

An important question is whether the hybridization signal intensity values correlate with the actual expression level of the transcript. Lockhart *et al.* [43] conducted a set of spike-in experiments (where the true transcript abundance is known) using oligo arrays to "determine the range of concentrations over which hybridization signals could be used for direct quantitation of RNA levels." They found that the hybridization intensity was linearly related to RNA concentration over a wide range of concentrations. The intensity was less than expected for higher concentrations because the probe sites were beginning to become saturated. Hence the hybridization time affects the linear response range.

Chudin *et al.* [15] discuss an experiment to assess the relationship between signal intensities and transcript concentration for Affymetrix GeneChip arrays. They indicate that previously, hybridization signal was shown to be proportional to actual transcript concentration using very specialized arrays with hundreds of distinct probe pairs per gene. They assessed these results using standard GeneChip arrays and under real world conditions. In order to do this, they considered spike-in hybridizations of four prokaryotic transcripts with and without fixed eukaryotic background. They observed a linear relationship between transcript abundance and signal intensity between 1 pM and 10 pM transcripts. Note that 100 pM indicates a transcript frequency of about 1 in 1,500. By comparing both PM and MM intensity to transcript abundance, it was demonstrated that MM probes are picking up signal and not just measuring nonspecific binding.

Later, we will consider the possibility that the relationship between signal and intensity might vary by array. We define the nature of this relationship using a calibration function.

2.8 Data Analysis and Modelling

For this stage of the microarray experiment it is assumed that the user has obtained an intensity measurement for each array at each spot. The data is usually background corrected (to adjust for optical noise) and normalized (to correct for systematic array differences). Most investigators are interested in determining the fold change for each probe set. The fold change represents the estimated relative gene expression for some condition as compared to a baseline. In addition, we would like to identify differentially expressed genes (those genes which are believed to exhibit different levels of expression when comparing two samples) and possibly do some cluster analysis. These issues will be discussed in later chapters.

2.9 An Experiment to Examine Sources of Variability in Microarrays

Bakay *et al.* [5] report on a series of microarray experiments to study sources of variability. They considered two samples of muscle biopsy from 28 different patients. Of these individuals, 15 were suffering from Duchenne muscular dystrophy and 13 were healthy controls. Gene expression data was obtained using the MuscleChip from Affymetrix.

To test for array variability, two different hybridization solutions were applied to duplicate arrays and the correlations were calculated. Correlation was high (0.96 or greater) for both of the replicate pairs they considered. To examine the effect of the conversion of RNA to cRNA, RNA from six different sources were examined with twelve U74Av2 GeneChips. For a given (mixed) sample, RNA was isolated, RNA samples split, and duplicate cDNA, cRNA, and hybridizations were separately performed. They once again found high correlations (R^2=0.99 for five of six samples, with average R^2=0.978). Based on these high correlations, the authors concluded that neither of these sources (array or hybridization effects) were major sources of variability.

They also considered within patient variability. For each patient in the study, the biopsy was split into two parts, RNA was isolated independently, and separately hybridized to an array. To compare two samples from the same individual the Affymetrix difference calls (increased, decreased and no change) were considered. The variance was quantified as the percentage of increased or decreased difference calls between two samples from the same patient. The results varied dramatically from 1.5% to 18% of the 4,601 probe sets studied. Hence, tissue heterogeneity (within subject variability) was found to be a considerable source of variation.

Note that the data collected by Bakay *et al.* is rich in information, but was not fully exploited by the researchers. It would have been interesting to model the data and estimate the variance due to different sources. However, the authors based

most of the conclusions on correlations and difference calls. Unfortunately, the data collected in this study is not publicly available.

Chapter 3

MODELS FOR MICROARRAY DATA

The goal of almost all microarray experiments is the estimation of the relative abundance of mRNA transcripts. In order to accomplish this, a model must be fit to the intensity data to account for sources of variation. The purpose of these models is to relate observed intensity to transcript abundance. Most authors recognize the need to preprocess the intensity data because of the presence of systematic errors.

To understand the preprocessing steps and the general forms of the models, we consider a conceptual microarray experiment. Let us define θ_{ijn} as the true mRNA abundance of gene n for the sample i hybridized to array ij. Then for kth PM probe in the probe set corresponding to gene n (referred to as PM probe kn), the amount of target mRNA hybridized can be expressed as $\Phi_{kn}\theta_{ijn}$. In other words, each PM probe will capture some proportion of the total possible mRNA. Hence Φ_{kn} represents the binding affinity for PM probe kn. $\Phi_{kn}\theta_{ijn}$ is the amount of target signal or gene specific binding (GSB). However, some cross hybridization or nonspecific binding (NSB) is likely to take place, so the amount of mRNA actually bound at the spot will be equal to $\Phi_{kn}\theta_{ijn} + \nu_{ikn}$, where ν represents nonspecific binding. All analyses actually start with intensity data. A calibration function relates the amount of hybridization at a spot to the intensity of that spot. This will be a monotonic function and may vary by array. In addition, there may be some background intensity due to optical noise. If we define f_{ij} as the calibration function and b_{ijkn} as the background value, then the intensity for PM probe kn will

be $b_{ijkn} + f_{ij}(\Phi_{kn}\theta_{ijn} + \nu_{ikn})$. So, the true target transcript abundance (θ_{in}) is related to the intensity (I) as

$$\theta_{ijn} \rightarrow \Phi_{kn}\theta_{ijn} \rightarrow \Phi_{kn}\theta_{ijn}+\nu_{ikn} \rightarrow f_{ij}(\Phi_{kn}\theta_{ijn}+\nu_{ikn}) \rightarrow b_{ijkn}+f_{ij}(\Phi_{kn}\theta_{ijn}+\nu_{ikn}) = I_{ijkn}.$$

We use the calibration function to relate hybridized signal to intensity. Normalization is the preprocessing step which allows for comparison across arrays. So normalization corrects for the calibration function. For oligo arrays, the normalization is performed between arrays. For cDNA arrays, the intensities of the two dyes on a single array must be normalized. Normalization is discussed in further detail in Chapter 6.

Most microarray data analysis methods provide an expression index that is indicative of transcript abundance. From this index an estimate of the relative abundance for a treatment versus control array is computed. We can also use the data to test whether or not a gene has been differentially expressed.

In this chapter, models for obtaining expression indices are presented for both oligo arrays and cDNA arrays. For oligo arrays there are many competing models. We will attempt to discuss the most commonly used of these in detail.

MAS: Microarray Suite (MAS) is the software developed by Affymetrix. Since Affymetrix GeneChips are the most commonly used oligo arrays, MAS is a commonly used analysis tool. Affymetrix chips employ PM probes to capture target signal and MM probes to account for nonspecific binding and background noise. In MAS 4.0 expression index was based on PM-MM values. This was a problem because MM $\geq$ PM for roughly 1/3 of probe pairs on an array; this leads to the possibility of negative expression values [33]. The expression index in MAS 5.0, the most recent version of MAS, is based on ideal mismatch (IM) corrected values. IM is based on the measure MM value but defined such that $IM < PM$ to prevent negative values. The model behind the analysis performed by MAS is not explicitly stated and the

exact algorithm can only be performed using Affymetrix Microarray Suite. In our discussion, we have extracted the MAS 5.0 model assumptions as best as we can based on available literature for purposes of comparison to other models.

MBEI: Li and Wong [41] were the first to propose model based expression indices (MBEI). Following the example set by MAS, Li and Wong originally modelled the PM-MM values. However, they later proposed a model using PM values only (probably influenced by Irizarry *et al.* [33]). Expression indices from either of these models can be obtained from the publicly available software dChip.

RMA: Irizarry *et al.* [33] proposed an expression index based on PM values only. Unlike the multiplicative model with additive error used by Li and Wong, Irizarry *et al.* suggest a multiplicative error. They call their expression index a robust multi-array average (RMA). The original RMA model does not properly account for nonspecific binding. However, the recently proposed GC-RMA [76] does account for nonspecific binding. Both versions of RMA can be carried out in Bioconductor. Bioconductor is R based and free to the public. In addition to RMA and GC-RMA, the approximate algorithms for MAS and MBEI (and many others) are also available in Bioconductor.

Other Models for Oligo Arrays: Chu *et al.* [14] propose a flexible mixed models approach to the analysis of microarray data. This analysis can be carried out in SAS. Rocke and Durbin [57] propose a model with both additive and multiplicative error terms. Efron *et al.* [19] employ an empirical Bayes model for detecting differentially expressed genes.

A note on first and second generation models for oligo arrays: Currently MAS 5.0, MBEI and RMA are the most commonly used models for oligo arrays. After these models, mixed models (in the style of Chu *et al.*) and the empirical Bayes

analysis are probably the next most common methods. All of these models take an empirical approach to account for nonspecific binding. These can be considered a first generation of models. First generation models are discussed here and in Chapters 4, 5, and 6.

More recently, there has been a better understanding of the theoretical issues involved in nonspecific binding. In 2003, two papers ([48],[79]) were published explaining why some probes have higher NSB than others. In December 2004, Wu *et al.* published a paper introducing GC-RMA (which builds on the results of the two earlier papers) to the statistics community. GC-RMA can be considered part of a second generation of models, taking a mechanistic approach to NSB. Second generation models are discussed here and in Chapter 7.

Models for cDNA Arrays: In addition to models for oligo arrays, we also discuss some models for cDNA arrays. Since cDNA arrays were in use before oligo arrays, the models for cDNA arrays have influenced the analysis of oligo arrays. We describe the models used by Kerr *et al.* [46] and Dudoit *et al.* [17].

In the following discussion (of oligo models), all models will use common subscripts for the kth probe pair of the nth probe set for the jth replicate of the ith treatment. Assume that there are K_n probe pairs in probe set n.

3.1 Affymetrix Microarray Suite (MAS)

Recall that Affymetrix GeneChips use PM and MM probes to represent different genes. The MM probe is used to account for nonspecific hybridization that may be affecting the PM probe. This can be done, for instance, by considering PM-MM values rather than PM values alone. In some cases, however, the MM value may be larger than the corresponding PM value. In these instances it doesn't seem appropriate to subtract MM from PM, so Affymetrix MAS 5.0 accounts for

background and nonspecific binding using what they call an "ideal mismatch" (IM) value.

The Affymetrix "Statistical Algorithms Description Document" [3] defines a signal log value for a probe set as:

$$SignalLogValue_{ijn} = T_{bi}(log_2(PM_{ij1n} - IM_{ij1n}), ..., log_2(PM_{ijK_nn} - IM_{ijK_nn}))$$

where $IM_{ijkn} = MM_{ijkn}$ if $MM_{ijkn} < PM_{ijkn}$ or $IM_{ijkn} = \frac{PM_{ijkn}}{2^s}$, where $s > 0$ (and $IM_{ijkn} < PM_{ijkn}$) otherwise. So the signal log value for probe set n is defined as the one-step Tukey biweight of $log_2(PM - IM)$ values. The Tukey biweight algorithm is a method to determine a robust average unaffected by outliers [47, p205].

The Affymetrix analysis implies the following underlying model:

$$log_2(PM_{ijkn} - IM_{ijkn}) = log_2(\theta_{ijn}) + \varepsilon_{ijkn},$$

where θ is some expression index proportional to the amount of hybridized signal. However, the array data is then scaled using the factor

$$sf_{ij} = \frac{Sc}{TrimMean(2^{SignalLogValue_{ijn}}, 0.02, 0.98)} = \frac{Sc}{\overline{\theta}_{ij\cdot}}$$

where Sc is the "target signal" (default value of Sc=500). Hence, the average expression value for each array will be equal to some target value (usually Sc=500). Considering this scale normalization, the model is better expressed as:

$$log_2(PM_{ijkn} - IM_{ijkn}) = log_2\left(\frac{Sc}{\overline{\theta}_{ij\cdot}}\theta_{ijn}\right) + \varepsilon_{ijkn}.$$

Also, since the Affymetrix manual states that "probe effects refer to the inherent differences in the hybridization efficiency from different probes...calculating the ratio of signal for the same probe on two different arrays effectively cancels the intrinsic affinity factor for that sequence". This suggests the following model:

$$log_2(PM_{ijkn} - IM_{ijkn}) = log_2\left(\frac{Sc}{\overline{\theta}_{ij\cdot}}\Phi_{kn}\theta_{ijn}\right) + \varepsilon_{ijkn},$$

where Φ accounts for the binding affinity (a probe effect).

On the original scale we have:

$$PM_{ijkn} = IM_{ijkn} + \left(\frac{Sc}{\bar{\theta}_{ij.}}\Phi_{kn}\theta_{ijn}\right)2^{(\varepsilon_{ijkn})}. \tag{3.1}$$

3.2 Model Based Expression Index (MBEI)

The model proposed by Li and Wong [41] requires normalized data. They describe an invariant set normalization procedure, which iteratively selects a set of PM probes (from those probe sets that are thought to be unchanged)[51]. A piecewise-linear running median curve is fitted and used as a normalization curve.

Li and Wong [41] then model both PM and MM normalized values:

$$N(MM_{ijkn}) = \nu_{kn} + \theta_{ijn}\alpha_{kn} + \varepsilon_{ijkn}$$

$$N(PM_{ijkn}) = \nu_{kn} + \theta_{ijn}\alpha_{kn} + \theta_{ijn}\delta_{kn} + \varepsilon'_{ijkn}$$

where ν_{kn} is the response of the kth probe pair due to nonspecific hybridization, θ_{ijn} is an expression index for probe set n for the jth array of the ith sample, α_{kn} is the rate of the MM response on the kth probe pair, δ_{kn} is the additional rate of increase in the corresponding PM response, and ε is a generic symbol for a random error. Note that they also allow for a PM-only model of similar form [42].

Hence,

$$MM_{ijkn} = N^{-1}(\nu_{kn} + \theta_{ijn}\alpha_{kn} + \varepsilon_{ijkn})$$

$$PM_{ijkn} = N^{-1}(\nu_{kn} + \theta_{ijn}(\alpha_{kn} + \delta_{kn}) + \varepsilon'_{ijkn})$$

Substituting $\phi_{kn} = \alpha_{kn}$ and $\Phi_{kn} = \alpha_{kn} + \delta_{kn}$, we have

$$MM_{ijkn} = N^{-1}(\nu_{kn} + \theta_{ijn}\phi_{kn} + \varepsilon_{ijkn}) \tag{3.2}$$

$$PM_{ijkn} = N^{-1}(\nu_{kn} + \theta_{ijn}\Phi_{kn} + \varepsilon'_{ijkn}). \tag{3.3}$$

3.3 Robust Multi-Array Average (RMA)

Irizarry *et al.* [33] propose a background correction motivated by a signal plus noise model for PM intensities. They model the PM values as: $PM_{ijkn} = b_{ijkn} + s_{ijkn}$, where s_{ijkn} represents signal and b_{ijkn} represents optical noise and nonspecific binding. (We will see later that this background term does not really capture signal from nonspecific binding.) Background corrected PM values are defined as $B(PM_{ijkn}) \equiv E(s_{ijkn} \mid PM_{ijkn})$. Imposing a strictly positive distribution on signal (s_{ijkn}) also implies that background (b_{ijkn}) is strictly positive. They assume s_{ijkn} is exponentially distributed (with mean α) and b_{ijkn} is normally distributed (with mean μ and variance σ^2). To avoid negative values, the normal distribution is truncated at zero. If an observed (PM) intensity is O, then the background adjustment is defined as follows:

$$E(s_{ijkn} \mid O = o) = a + b\frac{\phi(\frac{a}{b}) - \phi(\frac{o-a}{b})}{\Phi(\frac{a}{b}) + \Phi(\frac{o-a}{b}) - 1},$$

where $a = s - \mu - \sigma^2\alpha$ and $b = \sigma$ [10]. Note that here ϕ and Φ represent the standard normal density and distribution function respectively and not affinity factors!

Based on an examination of a number of normalization techniques, Irizarry *et al.* advocate the use of quantile normalization [11]. The goal of this normalization method is to make the distribution of the probe intensities the same for all arrays in the experiment.

After background correction and normalization, Irizarry *et al.* [33] suggest modelling the background corrected, normalized, log transformed PM values as:

$$log_2(N(B(PM_{ijkn}))) = \mu_{ijn} + \alpha_{kn} + \epsilon_{ijkn}$$

where α_{kn} is the probe affinity effect, μ_{ijn} represents the log scale expression level for array ij and ϵ_{ijkn} represents an iid error term with mean 0. μ_{ijn} is estimated

using median polish [47, p178] and is used as the log scale measure of expression. This estimate is called the robust multi-array average (RMA).

If we define $B(PM_{ijkn})$ as

$$B(PM_{ijkn}) = PM_{ijkn} - b_{ijkn},$$

then,

$$N(PM_{ijkn} - b_{ijkn}) = 2^{\mu_{ijn}} 2^{\alpha_{kn}} 2^{\varepsilon_{ijkn}}.$$

Substituting, $\Phi_{kn} = 2^{\alpha_{kn}}$ and $\theta_{ijn} = 2^{\mu_{ijn}}$ we have

$$PM_{ijkn} = b_{ijkn} + N^{-1}(\Phi_{kn}\theta_{ijn}2^{\varepsilon_{ijkn}}). \tag{3.4}$$

Using spike-in and dilution data (where expression values and differentially expressed probe sets are known), Irizarry *et al.* compared RMA to expression indices computed by MAS 5.0 and MBEI. When considering bias, variance and ability to detect differentially expressed genes, they found that RMA compared favorably to the other two methods [33].

It should be noted that Sasik *et al.* proposed a model almost identical to the RMA model (Equation 3.4). They suggest modelling the background corrected, normalized and log transformed values as,

$$log_2(N(PM_{ijkn} - b)) = \varphi_{kn} + \gamma_{ijn} + \varepsilon_{ijkn}.$$

So, on the original scale, substituting $\Phi_{kn} = 2^{\varphi_{kn}}$ and $\theta_{ijn} = 2^{\gamma_{ijn}}$ we have:

$$PM_{ijkn} = b + N^{-1}(\Phi_{kn}\theta_{ijn}2^{\varepsilon_{ijkn}}).$$

Sasik *et al.* also favor the quantile normalization technique. The main difference between the methods is that the background correction is constant here, unlike the RMA model. Sasik *et al.* propose using all MM and some PM probes to estimate background. A PM probe is considered a background probe if its value is "close" to

its corresponding MM probe. The value b is estimated as the mode of the background probes. If a background corrected PM value is negative, it is excluded from further analysis.

Recently, Wu *et al.* [76] proposed a modification to the RMA model. After noting that "RMA does not adjust well for nonspecific binding", they incorporate a nonspecific hybridization adjustment into the original RMA model. Information about the G-C content of the probes is used to perform this new background adjustment and the resulting analysis is called GC-RMA. The only difference between RMA and GC-RMA is the background correction step.

GC-RMA is built on the belief that the amount of nonspecific binding that occurs is related to the sequence of the probe. Since G and C nucleotides form stronger bonds, it seems reasonable to suspect that oligos with higher G-C content might be more susceptible to nonspecific hybridization. Following the lead of Naef and Magnasco [48], Wu *et al.* model probe affinity as the sum of position dependent base effects which are estimated using data where only nonspecific binding is known to occur. The fitted affinity terms (α) are used to describe nonspecific binding noise. The authors point out that "the advantage of the affinities over the MM is that they will not detect signal since they are pre-computed numbers" [76].

Wu *et al.* now assume:

$$PM_{ijkn} = O_{ij} + N_{ijkn} + S_{ijkn} \tag{3.5}$$

$$MM_{ijkn} = O_{ij} + N'_{ijkn} + \phi_{kn} S_{ijkn}, \tag{3.6}$$

where O represents optical noise, N represents nonspecific binding noise and S is a quantity proportional to RNA expression. They assume O follows a log-normal distribution and that $log_2(N)$ and $log_2(N')$ follow a bivariate-normal distribution with means μ_{PM} and μ_{MM} and variance σ^2 and correlation ρ constant across probes. Furthermore, they assume $\mu_{PM} \equiv h(\alpha_{PM})$ and $\mu_{MM} \equiv h(\alpha_{MM})$, where h is a smooth

function. O and N are assumed to be independent. The parameters μ_{PM}, μ_{MM}, ρ and σ^2 are estimated from the data. The background adjusted signal is defined as the predicted value of S given that PM and MM are observed and h, ρ, σ^2 and ϕ are known. When analyzing observed microarray data, the authors propose using $\phi = 0$. They also assume O is an array dependent constant and use $\hat{O}_{ij} = min_{kn}(PM_{ijkn}, MM_{ijkn}) - 1$. Two methods (maximum likelihood (MLE) and empirical Bayes (EB)) for estimation are discussed.

Wu et al. use a simulation study to compare RMA, GC-RMA and MAS 5.0. They found that although RMA is the most precise (based on standard deviation), accuracy is improved by using MAS 5.0 or GC-RMA(MLE). When comparing methods using spike-in data, GC-RMA (EB) was the most accurate, while all of the RMA methods outperformed MAS 5.0 for precision.

3.4 A Mixed Models Approach

Chu *et al.* [14] propose modelling microarray data using a mixed model. To adjust for overall array effects, they recommend centering the logged values so that they have mean 0.

They then recommend one of the following models:

Model I:

$$log_2(PM_{ijkn}) = \mu_{in} + \rho_{kn} + (\mu\rho)_{ikn} + \beta log_2(MM_{ijkn}) + A_{j(i)n} + \varepsilon_{ijkn}$$

Model II:

$$log_2(PM_{ijkn}) = \mu_{in} + \rho_{kn} + (\mu\rho)_{ikn} + A_{j(i)n} + \varepsilon_{ijkn}$$

Model III:

$$log_2(PM_{ijkn} - MM_{ijkn}) = \mu_{in} + \rho_{kn} + (\mu\rho)_{ikn} + A_{j(i)n} + \varepsilon_{ijkn}$$

Here μ_i is a treatment effect, ρ_{kn} is a probe effect, $(\mu\rho)_{ikn}$ is an treatment-probe interaction and $A_{j(i)n}$ is an probe set-array effect.

Let us work with Model I and incorporate the centering into the model:

$$log_2(PM_{ijkn}) - c_{ij} = \mu_{in} + \rho_{kn} + (\mu\rho)_{ikn} + \beta log_2(MM_{ijkn}) - \beta c_{ij} + A_{j(i)n} + \varepsilon_{ijkn}.$$

Converting to the original scale we have:

$$PM_{ijkn} = c_{ij}2^{\mu_{in}}2^{\rho_{kn}}2^{(\mu\rho)_{ikn}}MM_{ijkn}^{\beta}2^{\beta c_{ij}}2^{A_{j(i)n}}2^{\varepsilon_{ijkn}}.$$

Substituting $c_{ij}^{*} = c_{ij}2^{\beta c_{ij}}, \nu_{ikn} = 2^{(\mu\rho)_{ikn}}, \Phi_{kn} = 2^{\rho_{kn}}, \theta_{in} = 2^{\mu_{in}}$ and $\delta_{j(i)n} = A_{j(i)n}$ gives

Model I:

$$PM_{ijkn} = c_{ij}^{*}MM_{ijkn}^{\beta}\nu_{ikn}\Phi_{kn}\theta_{in}2^{\delta_{j(i)n}}2^{\epsilon_{ijkn}}. \tag{3.7}$$

Similarly we find,

Model II:

$$PM_{ijkn} = c_{ij}\nu_{ikn}\Phi_{kn}\theta_{in}2^{\delta_{j(i)n}}2^{\varepsilon_{ijkn}} \tag{3.8}$$

Model III:

$$PM_{ijkn} = MM_{ijkn} + c_{ij}^{\prime}\nu_{ikn}\Phi_{kn}\theta_{in}2^{\delta_{j(i)n}}2^{\varepsilon_{ijkn}}. \tag{3.9}$$

So for these models, θ_{in} is the expression index for treatment i and probe set n, Φ_{kn} is the probe affinity effect, ν_{ikn} and the MM term (if applicable) address nonspecific binding, $\delta_{j(i)n}$ is an error term indicating that the amount of available mRNA may not be identical within samples, and c_{ij}, c_{ij}^{*} and $c_{ij}^{\prime}$ define scale normalization factors.

3.5 Other Models for Oligo Arrays

Rocke and Durbin [57] propose the following two-component model:

$$y = \alpha + \mu e^{\eta} + \varepsilon \tag{3.10}$$

where y is the intensity measurement, μ is the expression level in arbitrary units, and α is the mean background (mean intensity of unexpressed probe sets). The first error term is $\varepsilon \sim N(0, \sigma_\varepsilon^2)$, which represents the background error and the second error term is $\eta \sim N(0, \sigma_\eta^2)$, which represents proportional error. In addition, Durbin *et al.* [18] present a variance stabilizing transformation for microarray data. Geller *et al.* [24] "provide a method for normalization of Affymetrix GeneChips simultaneous with the determination of the transformation, producing a data set without chip or slide effects but with constant variance and with symmetric errors". In other words, they normalize the arrays and also perform the variance stabilizing transformation.

Efron *et al.* [19] propose a nonparametric empirical Bayes model for detecting differentially expressed genes. Instead of a model for probe set expression, they suggest a series of data reductions. Inference is based on an empirical Bayes model and allows for simultaneous comparisons.

3.6 Models for cDNA Arrays

Kerr *et al.* [46] discuss the use of analysis of variance for a replicated cDNA experiment. They identify four main experimental factors: arrays (A), dyes (D), treated and control RNA varieties (V) and genes (G). Since they employed a Latin square design (a dye swap experiment with 2 varieties and 2 arrays) for the experiment under discussion, each of the possible 16 factorial effects is confounded with one other effect. Non-aliased effects are orthogonal to one another. They employ a "shift-log" transformation to account for differences in the dyes. This transformation uses a single parameter (per array) to account for differences in the dyes. They estimate a constant shift s_i for each array to minimize the absolute deviation from the median of $log_2(G + s_i) - log_2(R - s_i)$, where G indicates green signal intensities and R indicates red signal intensities.

Let y_{ijkgr} be the transformed value ($log_2(G + s_i)$ or $log_2(R - s_i)$) from array i, dye $j = 1, 2$, variety (or treatment) k and spot r of gene g. (Note the change in notation from the oligo array models.) They used the following model:

$$y_{ijkgr} = \mu + A_i + D_j + V_k + G_g + (AG)_{igr} + (VG)_{kg} + (DG)_{jg} + \epsilon_{ijkgr}.$$

where μ is the overall mean, A_i is the array effect, D_g is the dye effect, V_k is the variety effect, G_g is the gene effect (across other factors). The $(AG)_{igr}$ terms accounts for spot effects. The $(DG)_{jg}$ terms account for the gene specific dye effects (observed in their experiment). Finally, the $(VG)_{kg}$ terms represent the variety by gene interaction and are the effects of interest. They use a bootstrapping procedure to create error-bars for the relative gene expression between the two samples.

Dudoit *et al.* [17] discuss a within slide normalization approach which accounts for spatial and intensity dependent effects. They use a univariate test to identify differentially expressed genes, then correct for multiple testing by using adjusted p-values. They start by plotting the log intensity ratio $M = log_2 R/G$ versus the mean intensity ratio $A = log_2\sqrt{RG}$ yielding a so called MVA plot. From these graphs it can be seen that the log intensity ratio is dependent on the spot intensity A. Because of this, they perform a within print-tip group normalization using loess scatter plot smoothing:

$$log_2 R/G \rightarrow log_2 R/G - c_m(A) = log_2 k_m(A) R/G,$$

where $c_m(A)$ is the loess fit to the MVA plot for spots printed using the mth print-tip. They normalize on all genes considered, but point out that in some cases only housekeeping genes might be used.

In order to identify differentially expressed genes, the consider the following test statistic for gene g:

$$t_g = \frac{\bar{x}_{2g} - \bar{x}_{1g}}{\sqrt{s_{1g}^2/n_1 + s_{2g}^2/n_2}},$$

where $\bar{x}_{1g}$ and $\bar{x}_{2g}$ denote the average expression level of gene g in the n_1 and n_2 control and treatment hybridizations. Also, s^2_{1g} and s^2_{2g} represent the sample variances of gene g's expression in the control and treatment hybridizations. A large absolute t-statistic indicates that a gene has different expression levels under control and treatment conditions. Replication is required in order to use this test statistic. They do not assume that the t-statistics follow a t-distribution, but instead use permutation to estimate their distribution. They recommend using the Westfall and Young [73] step-down procedure to adjust for multiple comparisons.

Chapter 4

SIMARRAY FOR COMPARING METHODS

Although microarray technology is widely accepted in the scientific community, there is still debate about which analysis methods are recommended for routine applications. We compare three commonly used methods for the identification of differentially expressed genes for high density oligonucleotide microarrays. We consider Affymetrix Microarray Suite (MAS) 5.0, Li and Wong's model based expression index (MBEI) and Irizarry's robust multi-array average (RMA). We chose not to examine the mixed models proposed by Chu *et al.* because they are quite computationally intensive (even for SAS). Although our interest lies in the detection of differentially expressed genes, most methods usually advocate some choice of normalization as part of the fold change analysis.

Here we present a simulation study which allows a comparison of methods on realistic data, but with known fold changes. The effectiveness of the methods currently in use have thus far been only rarely tested and then often using only spike-in or dilution data where only a few genes are known to be differentially expressed. See literature review below for a discussion of some papers comparing analysis methods for oligo arrays. One data set is not enough for validation. A simulation study allows us to manipulate the data to study different scenarios while at the same time allowing us to "replicate" the microarray experiment enabling only selected sources of variability to affect the results- a virtual validation study. Simulation studies are an increasingly common tool for discrimination between methods in many areas of science. Our simulation attempts to mimic naturally occurring data and is based

on an observed data set. We start from a real data set to create a template and impose variation due to random error. This mimics a validation study where arrays would be physically generated. The simulation is based on a model that attempts to encompass all three methods (MAS, MBEI and RMA) and many different sources of variation.

A primary goal of the study is to examine the errors in the estimated fold changes (for each of the methods) over repeated application of the simulation. Another key point is the ability of the methods to detect differentially expressed genes. Although these methods are not designed as significance tests, we can examine their impact on detection of differentially expressed genes by using the same significance test for all methods. In the same sense, we consider the ability of the methods to control for error rates. In this case we focus on the false discovery rate (FDR).

4.1 Literature Review

It is important to note that different expression measures have already been compared in a number of papers. Here we present a summary of the comparisons and conclusions.

Irizarry *et al.* [32] compared MAS 5.0, dChip (MBEI) and RMA according to the following criteria "(i) the precision of the measures of expression, as estimated by standard deviations across replicate chips; (ii) the consistency of fold change estimates based on widely differing concentrations of target mRNA hybridized to the chip; (iii) the specificity and sensitivity of the measures' ability to detect differential expression, presented in terms of receiver operating characteristic (ROC) curves." They used a spike-in and dilution study from GeneLogic and a spike-in experiment from Affymetrix to perform their comparison. The dChip PM-only model was used. As a measure of precision, they computed the probe-set specific log_2 expression standard deviation across replicates and found that RMA had a smaller standard

deviation across expression levels while the methods had similar accuracy. In order to examine the consistency of the fold change estimates at different concentrations, they considered the correlation of fold change estimates for two different concentration groups and found that RMA had higher correlation than the other two methods. In order to create ROC curves, the number of false positives (non-spiked in genes with fold change estimates greater than a specified cut-off) and the number of true positive (spiked-in genes with fold change estimates greater than the same cut-off) were computed over a range of cut-off values. To construct an ROC curve, true positive rate (sensitivity) was plotted against the false positive rate (1-specificity). In these plots the RMA curves dominated the dChip and MAS curves, indicating "that the differential expression calls obtained with RMA have higher sensitivity and specificity than those obtained with the other two measures." Similar results were found when using test statistics to generate ROC curves. Note that these ROC curves leave something to be desired. They do not show the false positive or false negative rates as compared to known (or estimated) fold change. Hence it is difficult to compare how two methods perform at detecting differentially expressed genes for the same fold change.

Cope *et al.* [16] proposed a number of plots and summary statistics to compare summary methods. They provide a web-tool offering all of the proposed assessment criteria and offered a friendly competition for comparing proposed methods. All methods were applied to the same data, namely the spike-in and dilution data sets from GeneLogic. RMA, dChip and MAS 5.0 were compared in the paper and as part of the competition. It is difficult to define a "winner" for this competition because of the number of different assessments that were performed.

Rajagopalan [53] compared the performance of MAS 5.0, dChip and an error-modelling approach implemented in Rosetta Resolver based on a human Latin square data set from Affymetrix. This data includes 14 experiments over which each of the

transcript concentrations are varied between 0 and 1024 pM. This is the same spike-in data used by Irizarry *et al.* [32]. They considered the log-log plot of signal versus actual transcript abundance and found that that all methods had a slope less than the ideal slope of one, with Resolver coming closest to one. Another comparison was based on the accuracy of the methods in detecting 2-fold changes in the data. They computed ROC curves using five p-value cutoffs. The authors found MAS and Resolver to be superior to dChip in detecting differentially expressed genes.

Seo *et al.* [62] proposed using detection p-values from MAS 5.0 as a weighting function to improve the performance of expression summary methods. Five methods (including RMA and dChip) were studied with and without the proposed weighting method. The methods were tested using two large microarray data sets with different levels of confounding noise. Performance of the methods was judged by their ability to cluster samples into appropriate groups. The comparison indicated that the dChip (PM-MM) model with detection p-value weighting showed the best overall performance.

Barash *et al.* [7] used MAS 5.0 as a baseline and compared it with the dChip (MBEI) and RMA (implemented through RMAExpress) algorithms. Their comparison was based on eight "real life" Affymetrix Hu95 arrays. They considered the following methods for detecting differentially expressed genes: threshold number of misclassifications, the INFO score, t-test and Wilcoxon's rank sum test. According to the authors threshold number of misclassifications "is a non-parametric method that scores a gene by its ability to set a discriminative threshold between two groups of experiments" and "the INFO score measures the level of homogeneity when partitioning a gene's rank by a single threshold value." They used the dChip PM-only algorithm in their comparison. They found that "in 60% of genes, dChip had lower variation over replicates" than MAS while they found a 96% improvement when comparing RMA to MAS. They also considered the "overabundance of differentially

expressed genes" by comparing the number of genes with "low p-value scores, to what would be expected under the matching null hypothesis." They found a "mild increase in the set of informative genes when using dChip and a greater increase when using RMAExpress" regardless of the statistical criteria.

Rosati *et al.* [58] compared methods using six RG-U34 arrays with two different treatments in a realistic setting where differentially expressed genes were not known advance. Each of the methods (RMA, MAS 5.0 and dChip) was used to detect a group of differentially expressed genes and then genes identified as differentially expressed were confirmed or rejected by the use of real-time PCR. For all methods, some reduction of the data was performed based on the present/absent calls. For MAS two methods of detection were considered. The criteria for detection varied by method (depending on the output of the individual programs), but less than 30 genes were selected by any given method. The authors found that the RMA approach had a very low false positive rate but that its false negative rate was higher than MAS or dChip. The MAS program identified all of the true-positives tested but yielded a higher rate of false-positives. The dChip analysis yielded intermediate results.

In what appears to be the most detailed comparison of methods to date, Choe *et al.* [13] compared a number of different analysis methods using a "new" spike-in experiment. The data set has 1309 spike-in genes on six DrosGenome1 Affymetrix GeneChips . This is a huge number of spike-ins compared to previous spike in experiments! The authors were primarily concerned with maximizing the detection of genes that are truly differentially expressed. When comparing methods, they considered a number of possibilities at each step in the analysis: background correction, normalization, correction for nonspecific binding, expression summary and testing for differential expression. The analysis was carried out using Bioconductor. For the testing step, the authors considered the t-test, statistical analysis of microarrays (SAM) [69] and CyberT [6]; all were considered on the raw and log_2

scale. Based on ROC curves, the authors found that CyberT outperformed the other methods. Hence they chose that method for comparing all of the other analysis steps. Further comparisons were also based on ROC curves and the authors chose a "best" method at each step of the analysis. For background correction, the MAS method outperformed the RMA background correction. For normalization there was no clear preference between constant, invariant set (dChip), quantile (RMA) and loess normalization approaches. When considering how to correct for nonspecific binding, the PM-IM correction used by MAS 5.0 was preferred over PM-MM or PM-only methods. They also tried GC-RMA and PerfectMatch [79] and found that GC-RMA outperformed PerfectMatch but was slightly less effective than using PM-IM. For expression summary, RMA was chosen over MAS and MBEI. The authors also recommend an overall loess normalization performed at the probe set level "to center the log-fold changes around zero". Note that many of these findings are in contradiction to previous studies based on other spike-in experiments. The authors suggest that this may be due to the very small number of differentially expressed genes in those previous experiments.

4.2 Simulation Algorithm: SimArray

In order to compare the performance of each of the methods, a statistical simulation study was conducted in R using Bioconductor. The benefit of using a simulation study to evaluate the methods (instead of spike-in or dilution data) is that we have realistic data for which we know the "truth" and the process can be replicated as many times as desired.

Before discussing the simulation algorithm, we refer back to the conceptual microarray experiment discussed in Chapter 3. We allow that the available transcript abundances for replicate arrays may not be the same. For this simulation, we consider this to vary by some multiplicative error (η). Based on these assumptions, the

following general models for observed PM and MM are suggested for treatment i, array j, probe k and probe set n:

$$PM_{ijkn} = b_{ijkn} + f_{ij}(\Phi_{kn}\theta_{in}(1 + \eta_{ijkn}) + \nu_{ijkn})$$

$$MM_{ijkn} = b'_{ijkn} + f_{ij}(\phi_{kn}\theta_{in}(1 + \eta'_{ijkn}) + \nu'_{ijkn}).$$

Note that if the NSB (ν) was assumed to be zero, then the PM model would reduce to the RMA model (Equation 3.4). If an additive error was used instead of a multiplicative error and the background (b) was assumed to be zero, then these models would reduce to the MBEI models (Equations 3.2 and 3.3).

If we assume a scale calibration function, our models can be reexpressed as:

$$PM_{ijkn} = b_{ijkn} + f_{ij}\nu_{ijkn} + f_{ij}\Phi_{kn}\theta_{in}(1 + \eta_{ijkn})$$

$$MM_{ijkn} = b'_{ijkn} + f_{ij}\nu'_{ijkn} + f_{ij}\phi_{kn}\theta_{in}(1 + \eta'_{ijkn}).$$

Notice that if we assumed that the background and NSB were the same for both members of a probe pair, this model would represent assumptions made by MAS (namely that MM can be used to account for optical noise and nonspecific binding). If we average the optical background (b) and intensity due to nonspecific signal ($f_{ij}\nu$) over all probes on the array and represent the average with a single background term, our models reduce to:

$$PM_{ijkn} = b_{ij} + f_{ij}(\Phi_{kn}\theta_{in}(1 + \eta_{ijkn})) + \varepsilon_{ijkn} \tag{4.1}$$

$$MM_{ijkn} = b_{ij} + f_{ij}(\phi_{kn}\theta_{in}(1 + \eta'_{ijkn})) + \varepsilon'_{ijkn} \tag{4.2}$$

So, b is the average background intensity (resulting from both optical noise and nonspecific binding) and f is a scale calibration function, Φ and ϕ are the binding affinities for the PM and MM members of a probe pair, θ is the abundance for a given transcript and treatment and η and ε are multiplicative and additive errors.

In order to justify the assumption of a scale calibration function, we examine the plots of preprocessed (background corrected and normalized) PM values versus raw PM values using both the RMA and dChip methods for preprocessing. Two HGU95A GeneChip arrays (from a spike-in data set that we will discuss later in this chapter) were normalized and background corrected. The scatter plots of the preprocessed versus raw PM values for both RMA and dChip are shown in Figure 4.1. Although the normalization methods associated with RMA and dChip allow for nonlinear normalization functions, the fitted normalization function appears to be approximately linear in this case.

The simulation must begin with realistic baseline and experimental arrays. Each run of the simulation is focused on creating replicates from this truth. The following steps are used to create the baseline and experimental arrays.

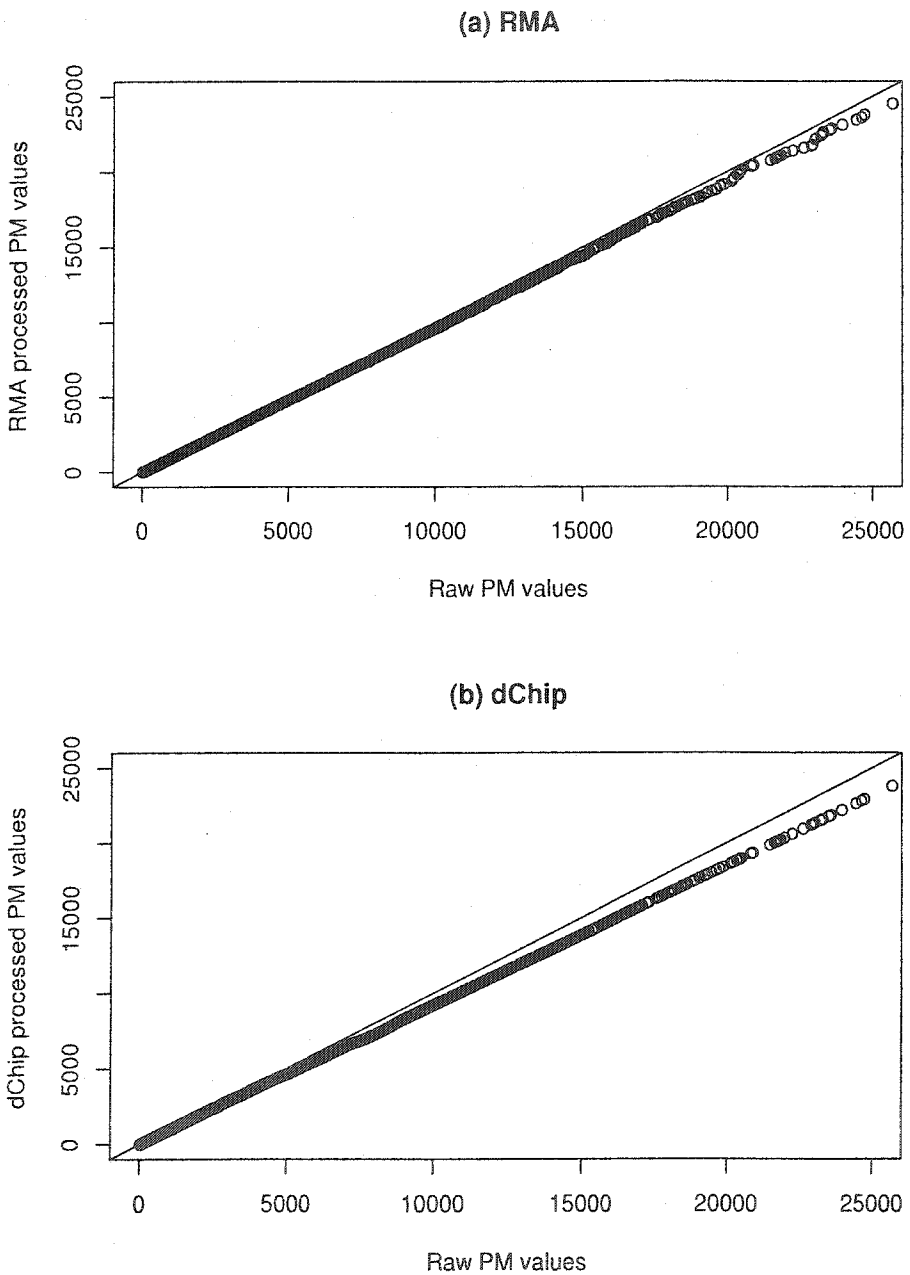
1. Choose a microarray study for which raw data are available. Select one array that is part of the study. This array will be used as the "template" for the simulation study.

2. Carry out the following data processing steps with the goal of decomposing the data nominally into signal and errors.

3. Subtract an estimated background intensity ($\hat{b_{ij}}$) defined as the minimum intensity of all probes. If we assume $f_{ij}(x) = 1$, then we are left with the signal values.

4. Scale the $(MM_{ijkn} - \hat{b_{ij}})/(PM_{ijkn} - \hat{b_{ij}})$ ratios by dividing by the 90^{th} percentile of the ratios. This forces 90% of the ratios to be less than or equal to one. This is justified by the belief that for most probe pairs the PM probe should have a higher binding affinity than the MM probe.

Figure 4.1: Scatter plots of preprocessed vs raw PM values for (a) RMA and (b) dChip.



5. Add variation to the probe values while maintaining the scaled ratios from step 4. To do this, we randomly choose one member of the probe pair (either PM or MM) and multiply by a random factor $(1+N)$ where N is normally distributed with mean 0 standard deviation 0.1. Then adjust the other member so that the scaled ratio is achieved. This will serve as the "true" baseline array.

6. To create the "true" experimental array, multiply all (PM and MM) probes for a given probe set by an appropriate fold change (FC) value. So, all probe pairs of the same probe set are defined to have the same fold change. We randomly assigned 2% of all probe sets to represent differentially expressed genes. For these probe sets, the fold changes were based on a gamma distribution. Recall that the gamma distribution allows for positive values only. For a given realization from the gamma distribution (x), the fold change was defined as either x+1 (for an up-regulated gene) or 1/(x+1) (for a down-regulated gene).

Now that we have a true baseline and experimental array, we generate replicate arrays using the simulation model (Equations 4.1 and 4.2). Hence we apply a multiplicative error (η), a calibration function (f) and additive error (ε). The background was generated from a uniform distribution and chosen so that all probe values were positive.

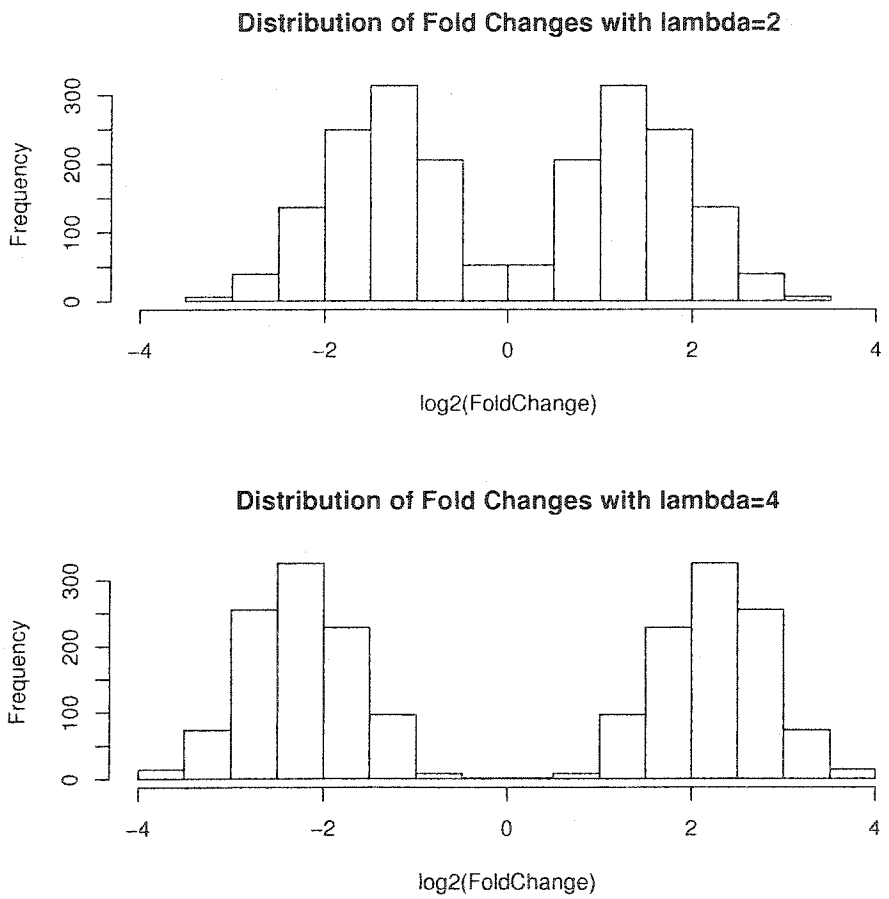
For a given run of the simulation the set of arrays is analyzed using MAS, RMA and MBEI (PM-only). For a given probe set, the estimated fold change is defined as the average expression for the treatment arrays divided by the average expression for the baseline arrays (where expression is recorded on the original- not log- scale). The raw p-value was based on a t-test using Welch-Satterthwaite degrees of freedom. These p-values were adjusted for multiple testing using the Benjamini-Hochberg method. Genes with an adjusted p-values less than 0.05 are declared differentially expressed. The fold changes and p-values were calculated for each probe set for each run of the simulation for each of the methods.

The program SimArray considers a number of different parameters. The number of replicates was either $r = 2$ or 5, for a total of four or ten arrays. For the probe sets representing differentially expressed genes, the fold changes were generated using a gamma distribution with a shape parameter of either $\lambda = 2$ or 4 and with a scale parameter of one. The distribution of the up-regulated fold changes for $\lambda = 2$ and 4 are shown in Figure 4.2. The multiplicative error (η) was generated using a normal distribution with mean zero and standard deviation of either $\sigma_\eta = 0.05$ or 0.10. The additive error (ε) was generated using a normal distribution with mean zero and standard deviation of either $\sigma_\varepsilon = 10$ or 20. Finally, we consider three calibration functions - the identity function, a scale calibration function and a nonlinear calibration function. We chose to examine the effect of a nonlinear calibration function because RMA and MBEI normalization methods allow for nonlinear normalization while MAS does not.

For this simulation study all runs for all scenarios start with data from a single array. The array data was taken from a spike-in experiment where 11 different cRNA fragments were added to the hybridization mixture of HGU95A GeneChip arrays at varying concentrations [28]. This data is available from GeneLogic at http://qolotus02.genelogic.com/datasets.nsf/. Array 92456hgu95a11 was used here. There are a total of 12,626 probe sets for this array. We assume that each probe set represents a gene.

It may seem risky to base all of our results on a single array. Remember that this is just a template for many replications of different scenarios of the simulation. We have attempted to account for as many sources of variation as possible. We are trying to mimic array to array variability. However, unknown differences across different arrays and types of arrays may exist. In order to account for this, we present the results of a limited additional study. The array data was taken from a set of experiments where normal and diabetic rats were treated with Vanadyl

Figure 4.2: Distribution of fold changes for differentially expressed genes with $\lambda = 2$ and 4.



Sulfate [74]. An Affymetrix RGU34A rat chip for a normal untreated rat was used as a template. There are a total of 8,799 probe sets for this array.

SimArray was programmed using R and Bioconductor. This allowed the use of the Bioconductor functions for each of the three methods (MAS, RMA and MBEI(PM-only)). Note that MBEI and MAS have their own software programs, but that they were not used for this simulation. In Bioconductor we used the following commands for the MBEI (PM-only) algorithm: expresso(data, normalize.method="invariantset", bg.correct=FALSE, pmcorrect.method="pmonly", summary.method="liwong"). Note that MBEI values returned from Bioconductor will differ from the values obtained from dChip. Using five replicates ($r = 5$) for each of the two treatments for the HGU95A template, the running time for 100 runs of the simulation was more than 24 hours.

4.3 Results

All results are based on 100 runs for a given simulation scenario. All methods use a stated false discovery rate of 0.05.

4.3.1 Fold Change Estimation

Most microarray users are interested estimating fold change between treatments. As a first look at fold change estimation errors, a set of scatter plots showing average estimated fold change (by method) versus true fold change for a single scenario ($\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with identity calibration function) are shown in Figure 4.3. This set of scatter plots is representative of many of the scenarios. The plots show that in a qualitative sense, the estimated fold changes appear to be reasonable for each method. For those genes which are differentially expressed, we consider the standard deviation of the $log_2(\hat{FC})$ values versus the absolute value of the true $log_2(FC)$ for the same scenario ($\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with identity calibration function) are shown in Figure 4.4. In order to further investigate

the errors in the estimated fold change, we calculated the $log_2(\hat{FC}_{method}/FC_{truth})$. Ideally we would like $\hat{FC}_{method} = FC_{truth}$ or $log_2(\hat{FC}_{method}/FC_{truth}) = 0$. We show box plots of this "logratio" by method for the HGU95A template in Figure 4.5 (for "unchanged" genes) and Figure 4.6 (for differentially expressed genes). This information is summarized in tabular form in Tables 4.1 and 4.2. The results for the RGU34A template are summarized in Tables 4.3 and 4.4. The results are discussed in Section 4.

Figure 4.3: Scatter plots showing average estimated fold change (by method) versus true fold change for the HGU95A template with $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with identity calibration function.

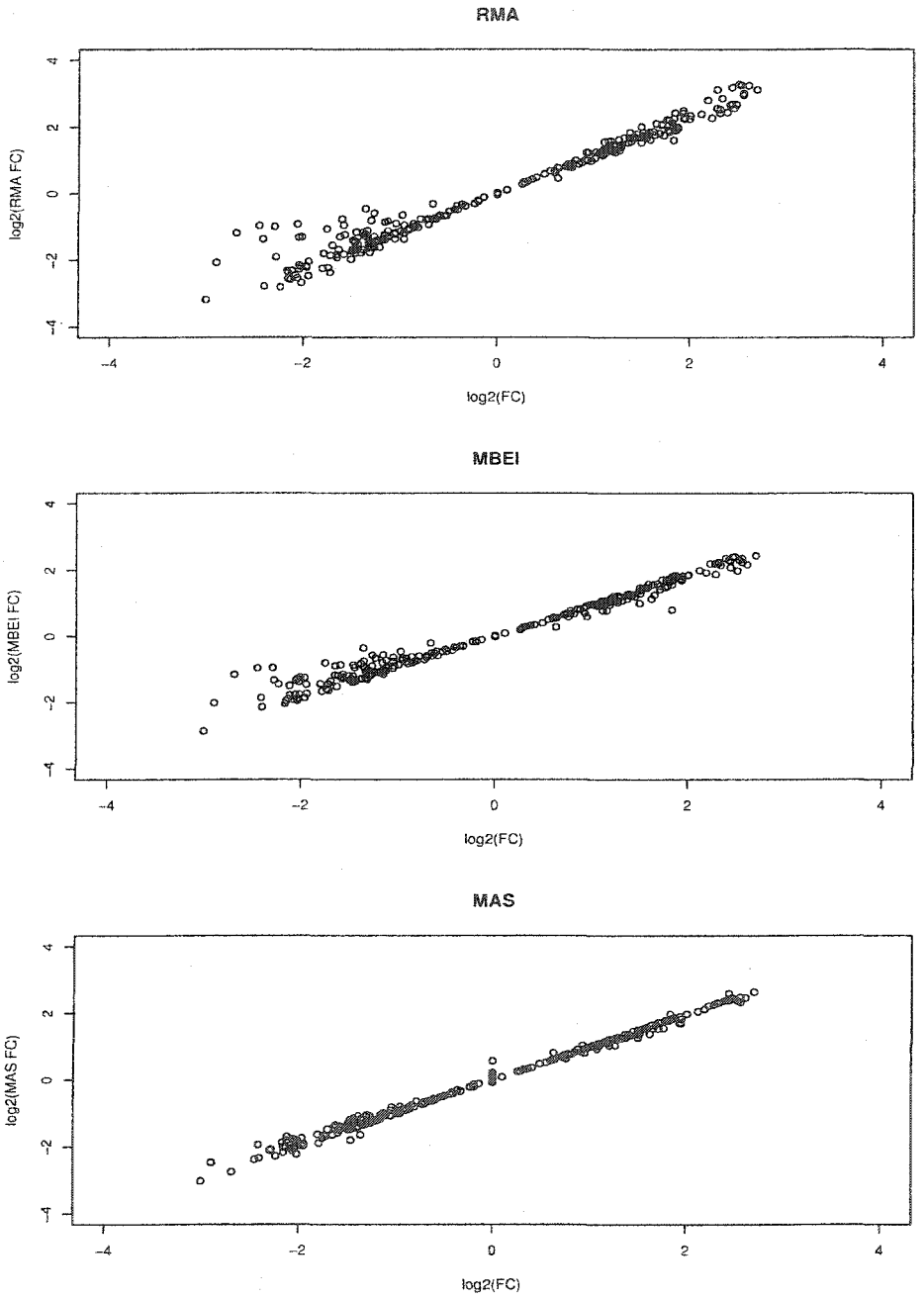


Figure 4.4: Scatter plots showing the standard deviation of the $log_2(\hat{FC})$ estimates (by method) versus the absolute value of the true $log_2(FC)$ for the HGU95A template with $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with identity calibration function.

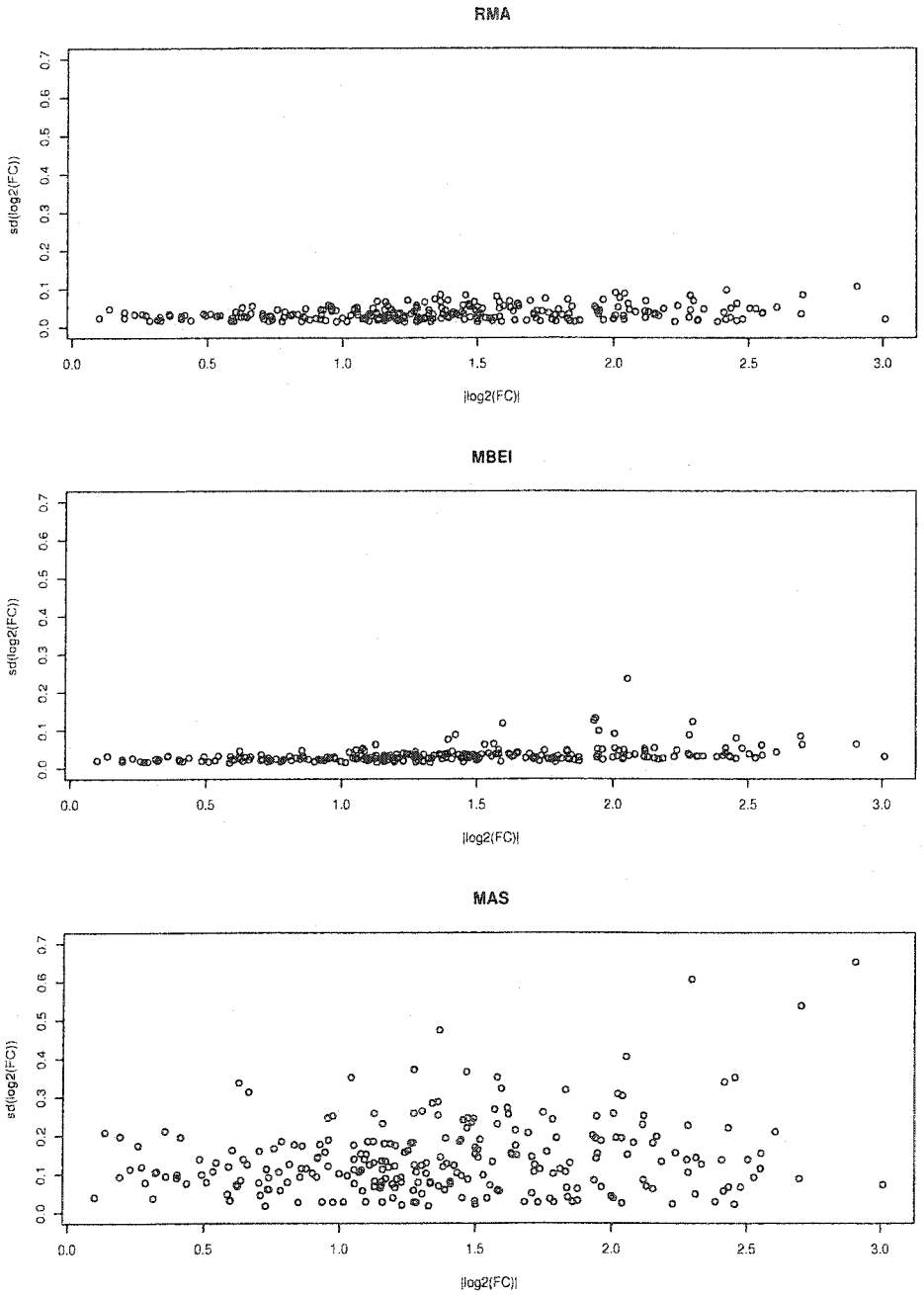


Figure 4.5: Box plots showing quantiles of $log_2(\hat{FC}_{method}/FC_{truth})$ for the HGU95A template by method for "unchanged" genes. (a) Box plots for $r = 5, \lambda = 2$ with low error and identity calibration function. (b) Box plots for $r = 5, \lambda = 2$ with high error and identity calibration function. (c) Box plots for $r = 5, \lambda = 4$ with low error and identity calibration function. (d) Box plots for $r = 5, \lambda = 4$ with high error and identity calibration function. (e) Box plots for $r = 5, \lambda = 2$ with low error and scale calibration function. (f) Box plots for $r = 5, \lambda = 2$ with low error and nonlinear calibration function. (g) Box plots for $r = 2, \lambda = 2$ with low error and identity calibration function. (h) Box plots for $r = 2, \lambda = 4$ with low error and identity calibration function.

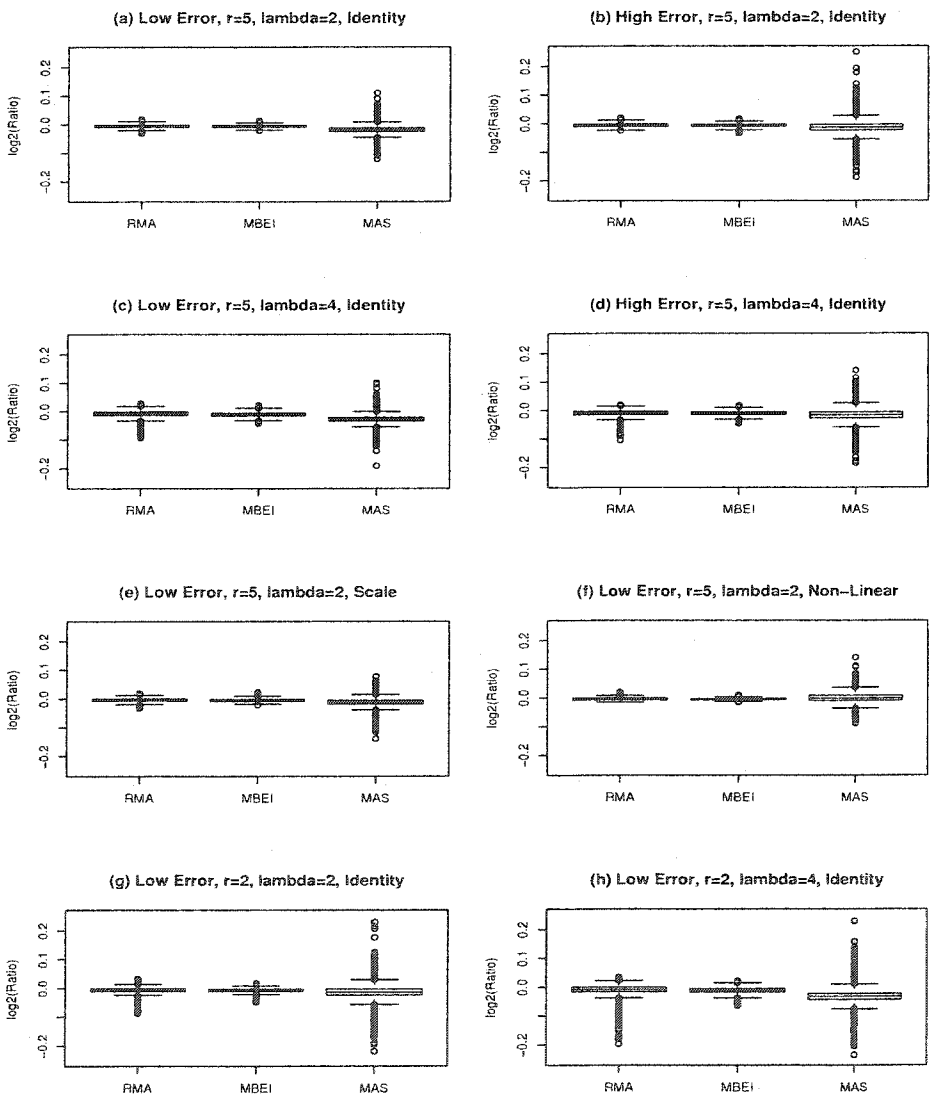


Figure 4.6: Box plots showing quantiles of $log_2(\hat{FC}_{method}/FC_{truth})$ for the HGU95A template by method for differentially expressed genes. (a) Box plots for $r = 5, \lambda = 2$ with low error and identity calibration function. (b) Box plots for $r = 5, \lambda = 2$ with high error and identity calibration function. (c) Box plots for $r = 5, \lambda = 4$ with low error and identity calibration function. (d) Box plots for $r = 5, \lambda = 4$ with high error and identity calibration function. (e) Box plots for $r = 5, \lambda = 2$ with low error and scale calibration function. (f) Box plots for $r = 5, \lambda = 2$ with low error and nonlinear calibration function. (g) Box plots for $r = 2, \lambda = 2$ with low error and identity calibration function. (h) Box plots for $r = 2, \lambda = 4$ with low error and identity calibration function.

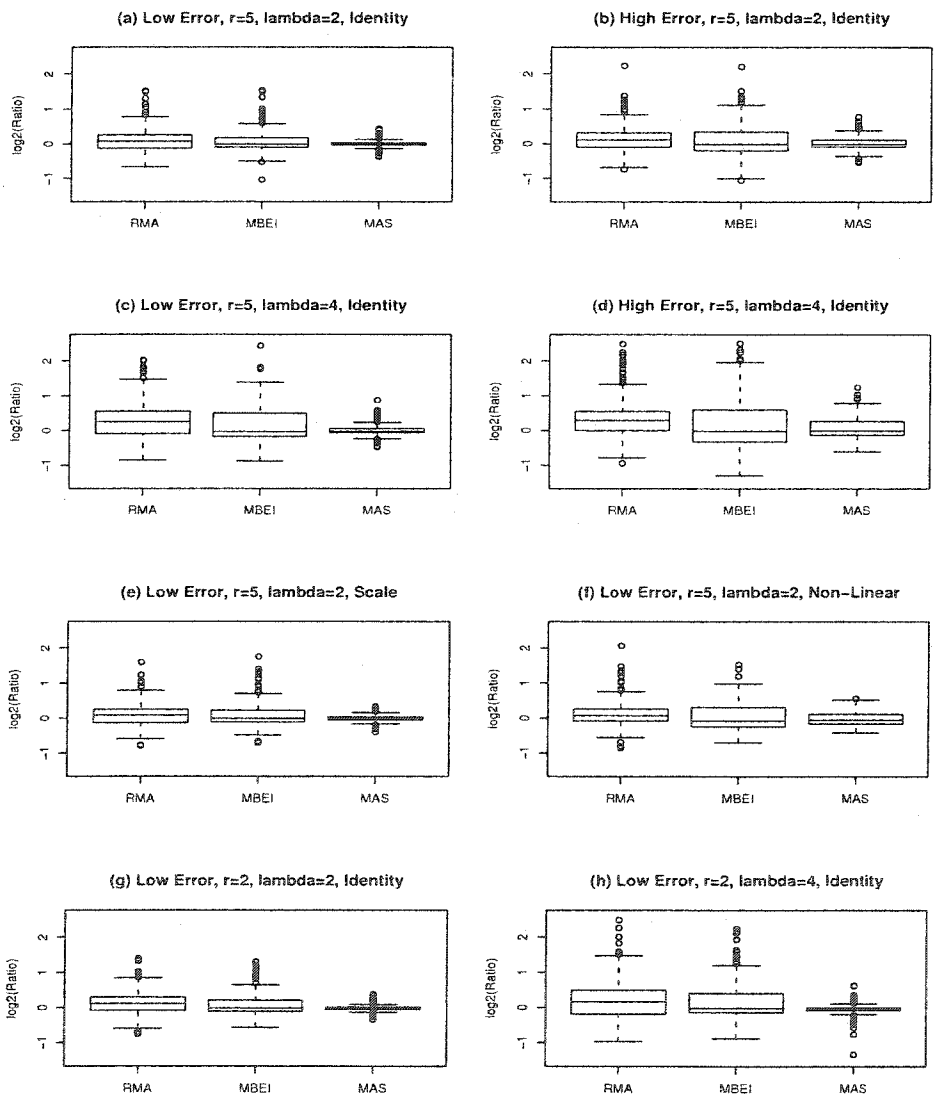


Table 4.1: Percentiles of $log_2(\hat{FC}_{method}/FC_{truth})$ for the HGU95A template for unchanged genes.

σ_η	σ_ε	r	λ	Calibration Function	Method	min	25%	50%	75%	max
0.05	10	5	2	Identity	RMA	-0.032	-0.009	-0.005	-0.001	0.019
					MBEI	-0.021	-0.010	-0.006	-0.003	0.014
					MAS	-0.121	-0.025	-0.018	-0.011	0.110
0.1	20	5	2	Identity	RMA	-0.025	-0.010	-0.006	-0.001	0.020
					MBEI	-0.034	-0.011	-0.007	-0.003	0.018
					MAS	-0.189	-0.024	-0.014	-0.003	0.249
0.05	10	5	4	Identity	RMA	-0.094	-0.015	-0.009	-0.002	0.028
					MBEI	-0.044	-0.017	-0.011	-0.006	0.022
					MAS	-0.191	-0.035	-0.028	-0.022	0.101
0.1	20	5	4	Identity	RMA	-0.105	-0.015	-0.009	-0.003	0.019
					MBEI	-0.047	-0.016	-0.011	-0.006	0.018
					MAS	-0.185	-0.028	-0.017	-0.006	0.141
0.05	10	5	2	Scale	RMA	-0.033	-0.008	-0.004	0.000	0.019
					MBEI	-0.023	-0.009	-0.006	-0.002	0.023
					MAS	-0.139	-0.019	-0.012	-0.006	0.078
0.05	10	5	2	Nonlinear	RMA	-0.016	-0.008	-0.005	-0.002	0.019
					MBEI	-0.016	-0.007	-0.005	-0.003	0.009
					MAS	-0.090	-0.010	-0.002	0.008	0.140
0.05	10	2	2	Identity	RMA	-0.088	-0.011	-0.007	-0.001	0.032
					MBEI	-0.048	-0.011	-0.008	-0.004	0.017
					MAS	-0.219	-0.025	-0.014	-0.004	0.229
0.05	10	2	4	Identity	RMA	-0.197	-0.015	-0.008	0.000	0.035
					MBEI	-0.065	-0.018	-0.012	-0.005	0.022
					MAS	-0.235	-0.043	-0.033	-0.022	0.228

Table 4.2: Percentiles of $log_2(\hat{FC}_{method}/FC_{truth})$ for the HGU95A template for differentially expressed genes.

σ_η	σ_ε	r	λ	Calibration Function	Method	min	25%	50%	75%	max
0.05	10	5	2	Identity	RMA	-0.667	-0.127	0.073	0.252	1.516
					MBEI	-1.037	-0.101	-0.007	0.176	1.536
					MAS	-0.378	-0.040	-0.016	0.029	0.439
0.1	20	5	2	Identity	RMA	-0.737	-0.089	0.121	0.321	2.235
					MBEI	-1.053	-0.190	-0.012	0.350	2.212
					MAS	-0.540	-0.079	-0.012	0.121	0.779
0.05	10	5	4	Identity	RMA	-0.849	-0.091	0.250	0.546	2.032
					MBEI	-0.884	-0.173	-0.033	0.490	2.442
					MAS	-0.478	-0.066	-0.028	0.053	0.869
0.1	20	5	4	Identity	RMA	-0.938	-0.002	0.292	0.548	2.495
					MBEI	-1.295	-0.321	-0.023	0.594	2.510
					MAS	-0.605	-0.126	-0.007	0.260	1.245
0.05	10	5	2	Scale	RMA	-0.789	-0.127	0.084	0.244	1.593
					MBEI	-0.707	-0.112	-0.005	0.220	1.753
					MAS	-0.394	-0.052	-0.015	0.036	0.331
0.05	10	5	2	Nonlinear	RMA	-0.851	-0.071	0.073	0.263	2.069
					MBEI	-0.700	-0.241	-0.076	0.315	1.530
					MAS	-0.414	-0.156	-0.051	0.119	0.561
0.05	10	2	2	Identity	RMA	-0.754	-0.083	0.113	0.290	1.391
					MBEI	-0.568	-0.109	-0.023	0.202	1.300
					MAS	-0.344	-0.052	-0.019	0.005	0.373
0.05	10	2	4	Identity	RMA	-0.966	-0.188	0.166	0.489	2.491
					MBEI	-0.881	-0.136	-0.022	0.406	2.244
					MAS	-1.334	-0.078	-0.038	0.001	0.625

Table 4.3: Percentiles of $log_2(\hat{FC}_{method}/FC_{truth})$ for RGU34A template for "unchanged" genes.

σ_η	σ_ε	r	λ	Calibration Function	Method	min	25%	50%	75%	max
0.05	10	5	2	Identity	RMA	-0.043	-0.006	-0.004	-0.002	0.017
					MBEI	-0.017	-0.005	-0.003	-0.002	0.007
					MAS	-0.154	-0.008	-0.002	0.003	0.137
0.05	10	5	4	Identity	RMA	-0.223	-0.013	-0.008	-0.003	0.021
					MBEI	-0.072	-0.014	-0.010	-0.006	0.021
					MAS	-0.223	-0.024	-0.018	-0.012	0.098

Table 4.4: Percentiles of $log_2(\hat{FC}_{method}/FC_{truth})$ for RGU34A template for differentially expressed genes.

σ_η	σ_ε	r	λ	Calibration Function	Method	min	25%	50%	75%	max
0.05	10	5	2	Identity	RMA	-0.707	-0.101	0.079	0.276	2.393
					MBEI	-1.085	-0.109	0.000	0.179	2.231
					MAS	-0.419	-0.035	-0.003	0.020	0.417
0.05	10	5	4	Identity	RMA	-0.627	-0.145	0.153	0.432	2.435
					MBEI	-1.241	-0.164	-0.036	0.379	2.556
					MAS	-0.850	-0.043	-0.019	0.014	0.359

4.3.2 Detection of Differentially Expressed Genes and Error Rates

We note again that the methods are for estimating transcript abundance, not for the detection of differentially expressed genes. However, we can examine the effect of each method on the detection of differentially expressed genes by using the same significance test for all methods.

In order to adjust for multiple comparisons, we have chosen to use the Benjamini-Hochberg adjusted p-value. The Benjamini-Hochberg method (step-up false discovery rate controlling procedure) is designed to control the false discovery rate (FDR) [8]. Details of the Benjamini-Hochberg method as well as other methods aimed at controlling FDR are given in a paper by Reiner *et al.* [55]. Genes with adjusted p-values less than 0.05 were declared differentially expressed. The procedure is designed to control the FDR at 0.05. The actual FDR will not necessarily equal 0.05. To examine this possibility we proceed as follows. For each run of the simulation, the FDR is estimated as the proportion of falsely rejected hypotheses (the proportion of genes with adjusted p-value less than 0.05 that are not differentially expressed). If no discoveries are made, then the FDR is defined to be zero.

We also evaluated the false positive rates, on a gene by gene basis (still after an adjustment for multiple comparisons). We call this the comparison wise error rate (CWER). An estimate of CWER is the proportion of times the adjusted p-value for an "unchanged" gene was less than 0.05 out of the 100 simulation replications.

A third error rate that may be of interest to the users of a Bonferroni adjustment is the family wise error rate (FWER). This is estimated by the proportion of the 100 simulation replications where at least one gene is falsely identified as differentially expressed. When using the Benjamini-Hochberg adjustment, the family wise error rate is not controlled, only the FDR is. Consequently one would expect FWER to be much higher than 0.05.

Although FDR is our primary focus, CWER and FWER were evaluated and reported for the benefit of users of these other criteria. In particular it is important to note that Benjamini-Hochberg adjustment is very different from a Bonferroni adjustment, as they have different objectives. The mean and standard deviation of the FDR, the FWER and maximum CWER for the HGU95A template are shown in Table 4.5. The empirical cdf functions for the comparison wise error rates for the HGU95A template are shown in Figure 4.7. The results for the RGU34A template are summarized in Table 4.6.

Note that a specific multiple testing adjustment is not prescribed by any of the methods considered in this paper (MAS, RMA, MBEI). However, a multiple testing adjustment is certainly an appropriate step in the detection of differentially expressed genes in a microarray experiment. Although the results will vary somewhat depending on the multiple testing adjustment used, we can still compare the relative performance of the methods for a single multiple testing adjustment.

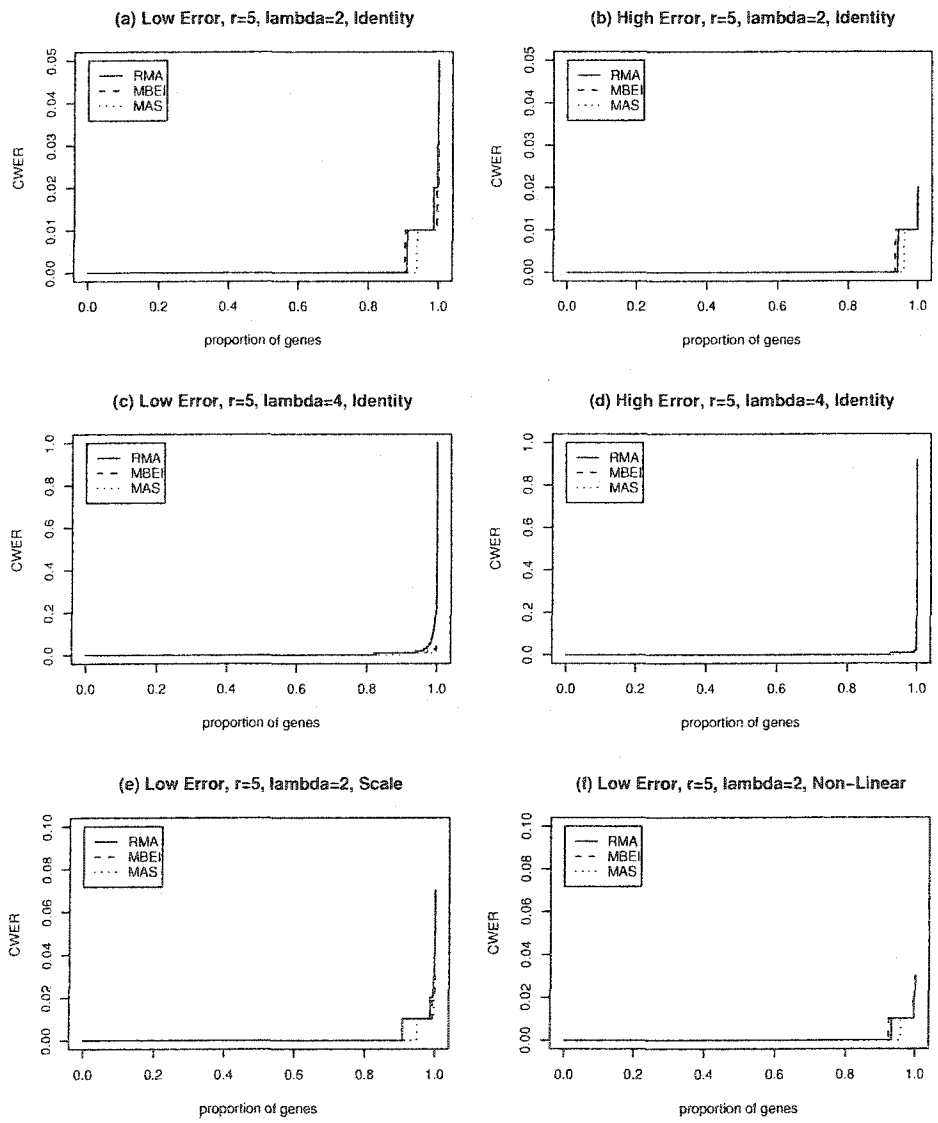
Table 4.5: Mean and standard deviation of false discovery rate (FDR), family wise error rate (FWER) and maximum comparison wise error rate (CWER) for the HGU95A template by scenario and method.

σ_η	σ_ε	r	λ	Calibration Function	Method	mean(FDR)	sd(FDR)	FWER	Maximum CWER
0.05	10	5	2	Identity	RMA	0.051	0.014	1	0.05
					MBEI	0.047	0.013	1	0.03
					MAS	0.038	0.013	1	0.03
0.1	20	5	2	Identity	RMA	0.029	0.010	0.99	0.02
					MBEI	0.033	0.012	1	0.02
					MAS	0.033	0.016	1	0.02
0.05	10	5	4	Identity	RMA	0.196	0.021	1	1
					RMA-IVSN	0.061	0.044	1	0.07
					MBEI	0.091	0.019	1	0.08
					MAS	0.056	0.015	1	0.04
0.1	20	5	4	Identity	RMA	0.045	0.014	1	0.92
					RMA-IVSN	0.036	0.044	1	0.03
					MBEI	0.042	0.014	1	0.03
					MAS	0.031	0.013	0.98	0.03
0.05	10	5	2	Scale	RMA	0.052	0.013	1	0.07
					MBEI	0.047	0.013	1	0.03
					MAS	0.033	0.014	1	0.02
0.05	10	5	2	Nonlinear	RMA	0.034	0.012	1	0.03
					MBEI	0.037	0.011	1	0.03
					MAS	0.029	0.027	0.96	0.02
0.05	10	2	2	Identity	RMA	0.019	0.114	0.04	0.01
					MBEI	0	0	0	0
					MAS	0.010	0.100	0.01	0.01
0.05	10	2	4	Identity	RMA	0.033	0.126	0.12	0.01
					MBEI	0.005	0.028	0.04	0.01
					MAS	0.005	0.050	0.01	0.01

Table 4.6: Mean and standard deviation of false discovery rate (FDR), family wise error rate (FWER) and maximum comparison wise error rate (CWER) for RGU34A template by scenario and method.

σ_η	σ_ε	r	λ	Calibration Function	Method	mean(FDR)	sd(FDR)	FWER	Maximum CWER
0.05	10	5	2	Identity	RMA	0.030	0.013	1	0.07
					MBEI	0.035	0.014	1	0.02
					MAS	0.029	0.014	0.99	0.03
0.05	10	5	4	Identity	RMA	0.248	0.024	1	1
					MBEI	0.119	0.023	1	0.46
					MAS	0.041	0.017	0.99	0.03

Figure 4.7: Empirical cdf functions for the comparison wise error rate (CWER) by method for the HGU95A template. (a) CWER empirical cdf functions for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with identity calibration function. (b) CWER empirical cdf functions for $\sigma_\eta = 0.10, \sigma_\varepsilon = 20, r = 5, \lambda = 2$ with identity calibration function. (c) CWER empirical cdf functions for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 4$ with identity calibration function. (d) CWER empirical cdf functions for $\sigma_\eta = 0.10, \sigma_\varepsilon = 20, r = 5, \lambda = 4$ with identity calibration function. (e) CWER empirical cdf functions for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with scale calibration function. (f) CWER empirical cdf functions for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with nonlinear calibration function.



4.3.3 Power

Power is defined as the probability that a test will declare a gene to be differentially expressed when in fact this is true. In other words, the power is equal to one minus the probability of a false negative. In order to estimate the power of the methods, we calculated, for each gene, the proportion of the 100 simulation replications where the adjusted p-value was less than 0.05. We expect this proportion to be small when a gene is not differentially expressed and large when the fold changes are large in either direction. We first plotted the estimated power as a function of $|log_2(FC)|$ for the genes in our data set. A logistic regression curve was fitted in order to smooth the plot. The fitted power curves for the HGU95A template are shown in Figures 4.8 and 4.9.

Figure 4.8: Power curves for the HGU95A template. (a) Power curves for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with identity calibration function. (b) Power curves for $\sigma_\eta = 0.10, \sigma_\varepsilon = 20, r = 5, \lambda = 2$ with identity calibration function. (c) Power curves for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 4$ with identity calibration function. (d) Power curves for $\sigma_\eta = 0.10, \sigma_\varepsilon = 20, r = 5, \lambda = 4$ with identity calibration function.

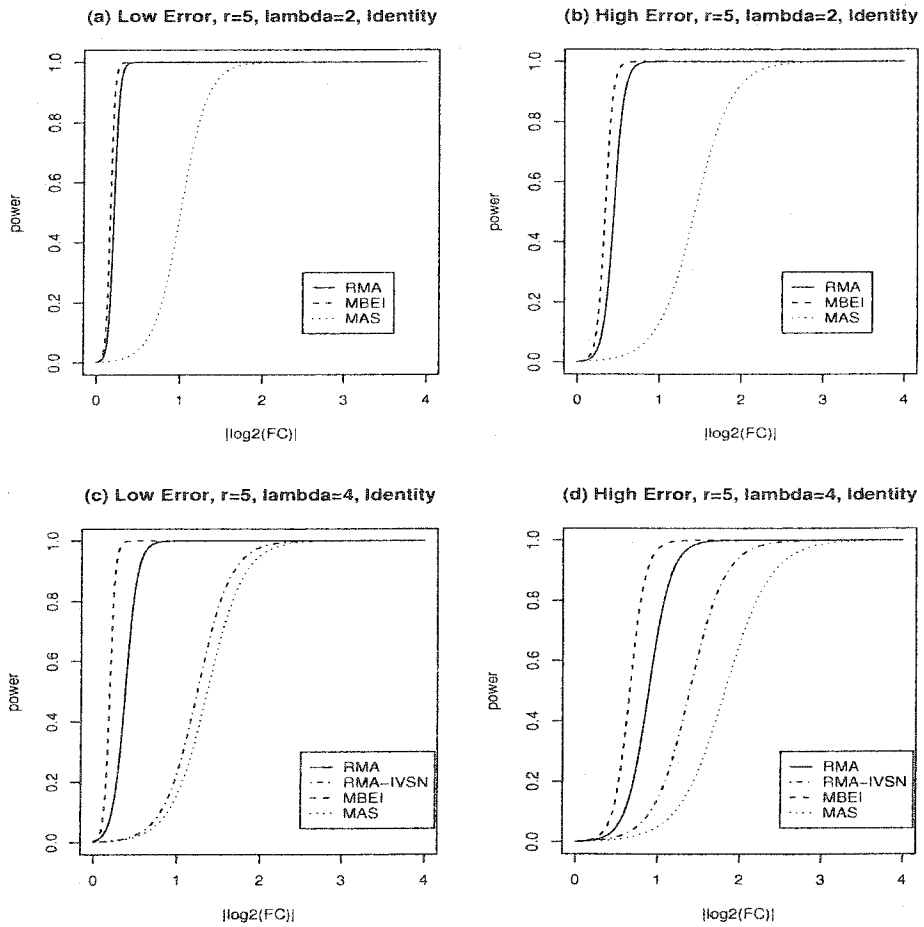
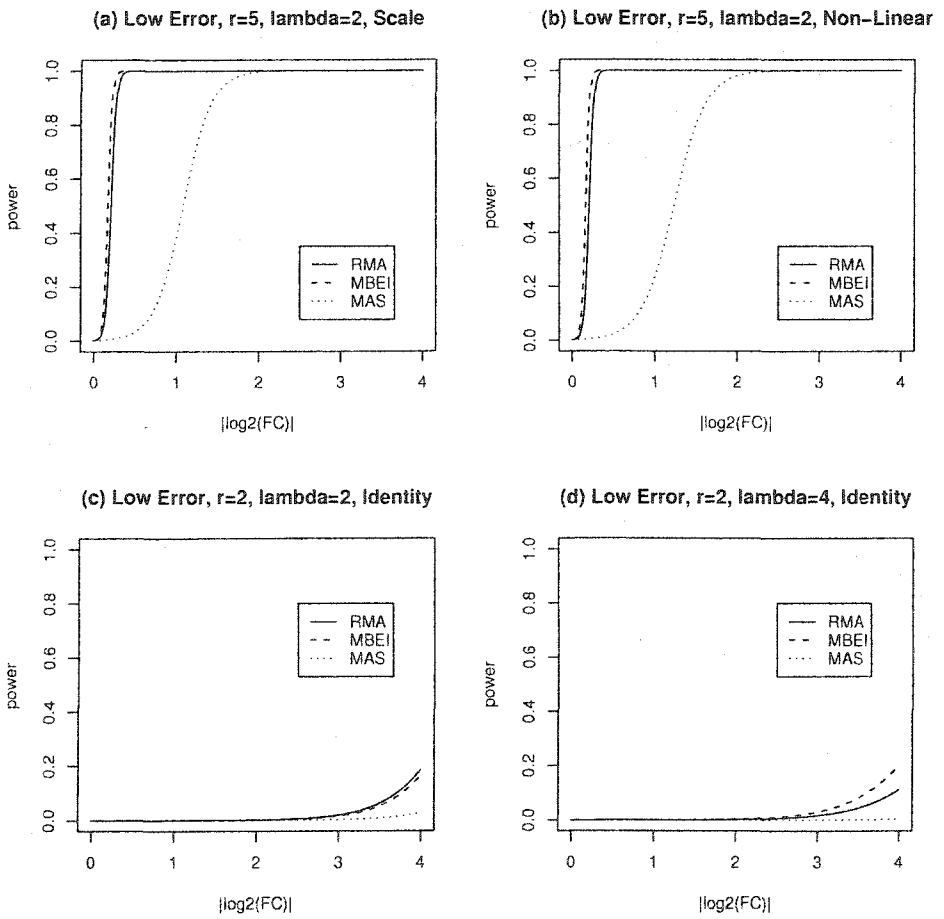


Figure 4.9: Power curves for the HGU95A template. (a) Power curves for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with scale calibration function. (b) Power curves for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 2$ with nonlinear calibration function. (c) Power curves for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 2, \lambda = 2$ with identity calibration function. (d) Power curves for $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 2, \lambda = 4$ with identity calibration function.



4.4 Discussion of Results

First we examine the errors in the fold change estimates for the HGU95A template. See Figures 4.5 and 4.6. Note that a 5% error in estimated fold change in either direction corresponds to a log_2-ratio of approximately ± 0.075. A 10% error in estimated fold change in either direction corresponds to a log_2-ratio of approximately ± 0.15. A 20% error in estimated fold change in either direction corresponds to a log_2-ratio of approximately ± 0.3.

For those genes which are unchanged, the errors in the estimated fold change are usually 5% or less. For all scenarios, MAS has the largest range of errors. Note that since the true fold change is one for a gene that is unchanged, the range of the estimated fold change (over all scenarios) is between 0.85 to 1.19. The RMA and MBEI methods appear to be performing similarly in most of the scenarios.

For those genes which are differentially expressed, the errors in the estimated fold change are considerably larger. For all scenarios, MAS has the smallest range of errors. The RMA and MBEI methods still appear to be performing similarly in most of the scenarios. Overall, it appears that the performance of MAS is more accurate for fold change estimation. Note that these results are supported by the limited results for the RGU34A chip (shown in Tables 4.3 and 4.4).

When we consider FDRs and FWERs for the HGU95A template shown in Table 4.5, we see that all methods reach their maximum mean(FDR) and CWER when $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 4$ with identity calibration function. For MAS, the maximum mean(FDR) = 0.056 and maximum CWER=0.04. For MBEI, the maximum mean(FDR) = 0.091 and maximum CWER=0.08. For RMA, the maximum mean(FDR) = 0.196 and maximum CWER=1. A closer look at the percentiles of the empirical cdf function for RMA when $\sigma_\eta = 0.05, \sigma_\varepsilon = 10, r = 5, \lambda = 4$ with identity calibration function show that the 99.9th percentile is 0.37.

This indicates that 0.1% of unchanged genes had a false rejection rate of 0.37 or higher. Similar results are seen for the RGU34A template (shown in Table 4.6).

Note that the RMA method appears to be competitive with MBEI in most scenarios. However, violations to the expected FDR and high CWER occur when $\lambda = 4$ and some larger fold changes are imposed. This led us to wonder if the normalization procedure might be at fault for these violations. In order to investigate this possibility, we reran the simulation using invariant set normalization (which is used by MBEI) with RMA expression index for scenarios with $\lambda = 4$. This hybrid version of RMA is denoted as RMA-IVSN in Table 4.5. Using RMA with IVS normalization does serve to reduce the maximum CWER as well as the mean(FDR). This indicates that the RMA normalization (quantile normalization) may be responsible for the inflated error rates for some scenarios. We discuss normalization algorithms in detail in Chapter 6.

Finally we consider the power curves for the HGU95A template as shown in Figures 4.8 and 4.9. For a given scenario, MAS always has the uniformly lowest power. This is probably due increased variance for this method (compared to RMA or MBEI) which can be seen in Figure 4.4. For all scenarios with $r = 5$, MBEI has power greater than or equal to that of RMA. Notice that RMA-IVSN has lower power than either RMA or MBEI. For all methods the power is quite low when there are only two replicates ($r = 2$). Reducing sample size will reduce the power. However, we expect that even experiments with small sample size would be able to detect genes with very large fold changes.

Notice that the multiplicative error terms for a probe pair (η and η') were assumed to be uncorrelated in the simulation model (Equations 4.1 and 4.2). We considered some limited scenarios where a second error term was used to add correlation between members of a probe set. For these scenarios, the simulated data was generated using the following model:

$$PM_{ijkn} = b_{ij} + f_{ij}(\Phi_{kn}\theta_{in}(1 + \eta_{ijkn})(1 + \delta_{j(i)n})) + \varepsilon_{ijkn}$$

$$MM_{ijkn} = b_{ij} + f_{ij}(\phi_{kn}\theta_{in}(1 + \eta'_{ijkn})(1 + \delta_{j(i)n})) + \varepsilon'_{ijkn}$$

where $\delta_{j(i)n}$ was generated using a normal distribution with mean zero and standard deviation of $\sigma_\delta = 0.05$ and $\sigma_\eta = 0.01$. We note here that even when this correlated error structure was employed, it did not appear to alter the conclusions of this study.

In conclusion, the MBEI expression index with corresponding normalization method is recommended because it appears to maintain its stated FDR while operating with high power.

Chapter 5

SIMARRAY FOR SAMPLE SIZE CALCULATIONS

Cost indicates that microarrays must still be used conservatively. Many investigators are interested in determining the number of arrays per treatment required to attain a certain power. However, because of the numerous steps involved in a microarray experiment (background correction, normalization, summary method, significance testing and multiple testing adjustment) this question is not trivial to answer.

Here we present an important use for the general SimArray framework - a simulation based sample size calculator which allows for estimation of power and false discovery rate. The required input includes one or two "starter" arrays (possibly taken from a previous experiment), a list of proposed fold changes and variance components estimates. We discuss how to choose these values. The user must also choose a stated model (either RMA or MBEI). From this initial input, SimArray simulates microarray data for a requested number of replicates from which the power (to detect differentially expressed genes) and FDR can be estimated. We illustrate these methods using barley data.

Our simulation attempts to mimic naturally occurring data and is based on one or two "starter" arrays. From a single "starter" array we can generate variation due to random error according to either the RMA or MBEI models. We choose to focus on RMA and MBEI algorithms because their models are explicitly stated in published works. In contrast we chose not to examine the MAS algorithm, because no

model is clearly stated for this algorithm. In addition, some details of the algorithm are not publicly available.

Other authors have examined sample size calculation for microarray experiments. Our approach is probably most similar to that of Zien *et al.* [81] who also take a simulation based approach and consider two sources of variation. However, their simulation occurs at the probe set level instead of the probe level and hence cannot consider the effects of normalization or even expression summary (i.e. RMA or MBEI). In addition, they do not consider the FDR. Yang *et al.* [77] discuss the number of subjects required to minimize the FDR while achieving high power to detect differentially expressed genes. They consider designs to (1) compare two groups at a single time point and (2) compare two experimental groups across time points. For the second design, they employ a simulation based method at the gene level without considering the method of expression summary. Black and Doerge [9] use the t-test and bootstrap methods (with a Bonferroni correction) when calculating the power for cDNA microarrays. Pan *et al.* [50] discussed sample size based on a mixture model. Hwang *et al.* [31] consider determining the sample size for finding the best discriminant function. Hence they were concerned with gene classification instead of detection of differentially expressed genes. Lee and Whitmore [39] consider a number of approaches for calculating sample size in cDNA microarray studies (although results could be applied to oligo array studies) including classical and Bayesian approaches.

5.1 Algorithm for SimArray as a Sample Size Calculator

SimArray is a program for calculating sample size by simulation. The user must begin with one or two "starter arrays", a list of fold changes and estimates for the variance component(s). Discussion about how we obtain reasonable values for the fold changes and variance components can be found in Section 5.2. In addition the

user must specify the method of expression summary (RMA or MBEI), the type of significance test (t-test or Wilcoxon rank sum), and the multiple testing procedure. After this information is provided, SimArray can estimate the power and error rates for various sample sizes.

SimArray is programmed in R using Bioconductor. This allows us to utilize many of the built in data analysis commands available from both programs. Of course, both programs are free and available to the public.

The simulation must begin with at least one "starter" array. If no observed data is available from the current or proposed study, a baseline array can be chosen from a previously completed study using the same type of array. Each run of the simulation is focused on creating replicates from this truth.

Ideally, one would like to simulate data according to a unified model representing truth. (We present a unified model in Chapter 7.) However, choice of parameters and selection of reasonable parameter values for a unified model is a difficult task. We chose to use a simplified method for simulation. We generate data by assuming the form of the RMA or MBEI models.

Below the SimArray algorithm for sample size calculation is outlined.

1. Start with a list of fold changes and a baseline array.

2. The baseline array should be preprocessed to correct for optical noise and nonspecific binding. If an experimental array is also available, then the data should be normalized. Preprocessing should be done according to either the RMA or MBEI algorithms. Background adjustments should be saved.

3. If we assume that the preprocessing has corrected for systematic errors and the error terms in either RMA model (Equation 3.4) or MBEI model (Equations 3.2 or 3.3) are zero, then we are left with signal values.

4. Create a "true" experimental array by multiplying the "true" baseline array by the assumed fold changes. All probe pairs within a probe set are defined to have the same fold change.

5. Generate replicate arrays according to either the RMA or MBEI algorithms. Details for RMA are given in Section 5.1.1. Details for MBEI are given in Section 5.1.2.

6. Estimate fold change and obtain p-values for differential expression using the chosen model (RMA or MBEI) and testing combination of interest. The raw p-value can be based on a t-test or Wilcoxon rank sum test. The p-values can be adjusted for multiple testing using the Benjamini-Hochberg method or any other multiple testing procedure. The fold changes and p-value should be calculated for each probe set for each run of the simulation.

7. Summarize performance by considering false discovery rate, power and appropriate graphs. We illustrate these summary values and graphs in Section 5.3.

5.1.1 Simulated Data for RMA Analysis

We start with the RMA model (Equation 3.4), however we consider the possibility that there may be an additional error term representing the array within treatment error $\delta_{j(i)n}$. This term would represent the fact that the amount of transcript abundance could vary from array to array for a single treatment. This is true for either biological or technical replicates.

So, for simulation purposes we use the following model:

$$PM_{ijkn} = b_{ijkn} + f_{ij}(\Phi_{kn}\theta_{in}2^{\epsilon_{ijkn}}2^{\delta_{j(i)n}}). \tag{5.1}$$

Working from the "true" baseline and experimental array, we generate replicate arrays using the following steps.

1. To generate the replicate arrays (for each run of the simulation), multiply by the error terms $2^{\delta_{j(i)n}}$ and $2^{\epsilon_{ijkn}}$, where $\delta_{j(i)n} \sim N(0, \sigma_\delta^2)$ and $\epsilon_{ijkn} \sim N(0, \sigma_\epsilon^2)$. The values σ_δ^2 and σ_ϵ^2 are user specified. When starter arrays are available, reasonable values to use for σ_δ^2 and σ_ϵ^2 may be deduced from starter array data. In Section 5.2.3, we explain the procedure we used for the barley data.

2. The f_{ij} terms are assumed known. For the barley example, we assumed a scale calibration function and generated scaling constants for each array independently from a uniform(0.9,1.1) distribution.

3. Background b_{ijkn} is added back to each array. The background terms for the barley example are discussed in Section 5.2.2.

5.1.2 Simulated Data for MBEI Analysis

Despite the MBEI model (Equations 3.3 and 3.2), a multiplicative error term better represents better represents the observed data. This view is supported by the form of the models used by other authors ([14],[33] and [57]). So for simulation purposes, we use the following model:

$$MM_{ijkn} = b_{ij} + f_{ij}(\theta_{in}\phi_{kn}(1 + \epsilon_{ijkn})) \tag{5.2}$$

$$PM_{ijkn} = b_{ij} + f_{ij}(\theta_{in}\Phi_{kn}(1 + \epsilon'_{ijkn})). \tag{5.3}$$

Working from the "true" baseline and experimental array, we generate replicate arrays using the following steps.

1. To generate the replicate arrays (for each run of the simulation), multiply by the error term $(1 + \epsilon_{ijkn})$, where $\epsilon_{ijkn} \sim N(0, \sigma_\epsilon^2)$ and σ_ϵ^2 is user specified. When starter arrays are available, a reasonable value to use for σ_ϵ^2 may be deduced from starter array data. In Section 5.2.4, we explain the procedure we used for the barley data.

2. The f_{ij} terms are assumed known. For the barley example, we assumed a scale calibration function and generated scaling constants for each array independently from a uniform(0.9,1.1) distribution.

3. Background b_{ij} is added back to each array. The background terms for the barley example are discussed in Section 5.2.2.

5.2 Illustration using Barley Data

To illustrate our sample size simulation methods, we apply our methods to some data from an experiment using Barley1 GeneChips. Samples of a number of cultivars were obtained at various stages of the malting process. Samples were hybridized to Affymetrix Barley1 arrays. Each Barley1 array contains 22,840 probe sets. Most probe sets have 11 probe pairs. We assume each probe set represents a gene. For our "starter arrays" we chose a sample of the Merit and Legacy cultivars each obtained on day 4 of the malting process. For MBEI, preprocessing was performed in dChip and data was later imported into Bioconductor for use in SimArray.

5.2.1 Estimating Fold Changes

For clarity, let us define the fold change as the ratio of transcript abundance between two samples for a given gene. To start, let us assume that both a baseline and experimental array are available. These arrays should be normalized and background corrected (by either the RMA or MBEI algorithm). If this is the case, the following estimation methods are possible:

1. As a naive estimate, we could estimate the fold change for probe set n as

$$\hat{FC}_n = \sum_i \frac{PM_{e.kn}}{PM_{b.kn}} / n_k$$

where $PM_{e.kn}$ is the PM value for the kth probe of the nth probe set on the baseline array and $PM_{b.kn}$ is the corresponding PM value on the baseline

array and n_k is the number of probe pairs in probe set n. So, one could use the average of the ratios of the PM values. Similarly we could use the median of the probe level ratios.

2. A regression based estimated can be obtained by fitting the model

$$PM_{e.kn} = b_n \times PM_{b.kn} + \epsilon_{kn}.$$

Hence $\hat{b}_n$ is an estimate of the fold change for the nth probe set. So, we fit a regression line (with zero intercept) for each probe set and use the slope as an estimate of fold change.

3. Another possibility is to perform the regression on the log_2 scale and use the exponential of the intercept as an estimate of the slope:

$$log_2(PM_{e.kn}) = a_n + b_n \times log_2(PM_{b.kn}) + \epsilon_{kn}.$$

In order for $exp(\hat{a}_n)$ to be an estimate of fold change (defined as the ratio of abundances), the value b_n must be equal to 1. In practice we found the slope estimates to vary considerably. If we constrain $b_n = 1$, then we estimate the intercept as

$$\hat{a_n} = \sum_i log_2(\frac{PM_{e.kn}}{PM_{b.kn}})/n_k$$

and the estimate of the fold change is $exp(\hat{a}_n)$.

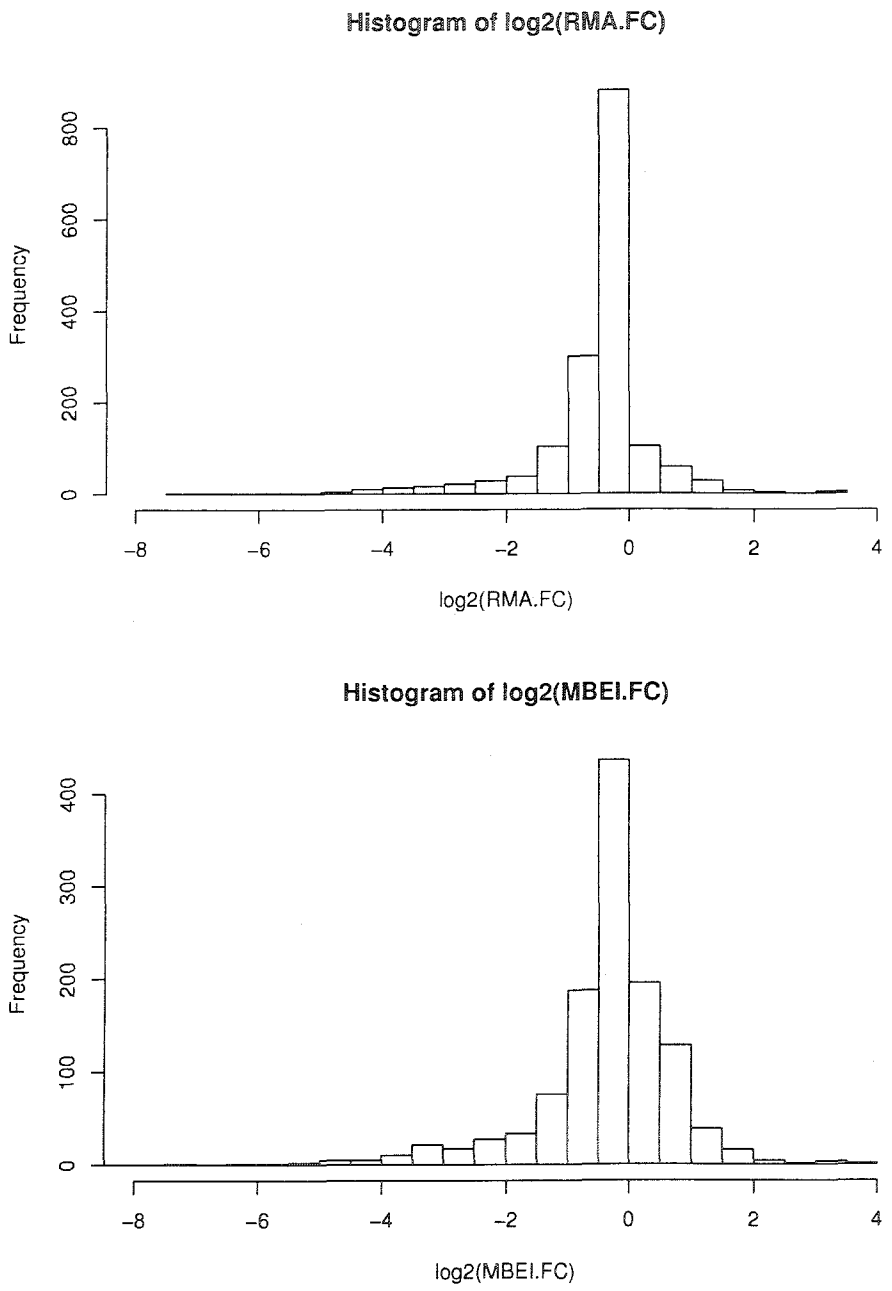
4. Yet another possibility includes using errors in variables regression estimates which allows for errors in both variables. This is reasonable because we expect errors in both PM_b and PM_e.

All of the estimates above have standard errors and significance testing can be performed to test for differentially expressed genes (from only a pair of arrays). In practice, a multiple testing adjustment should be used.

It is not possible to estimate fold change from a single array. If only a single array is available, fold changes can be generated by presuming some percentage of probe sets to represent differentially expressed genes with a given distribution. Alternatively, fold changes could be estimated from previous similar experiments.

For the barley data (for both RMA and MBEI), fold change estimates were obtained using the regression estimation method. Differentially expressed genes were identified by testing: $H_0 : FC_n = 1$ versus $H_a : FC_n \neq 1$. Initial p-values were based on the t-test, but later adjusted for multiple comparisons using the Benjamini-Hochberg adjustment. For RMA, approximately 7% of genes (on the array) are considered to be differentially expressed. For MBEI, approximately 5% of genes (on the array) are considered to be differentially expressed. The distribution of the log_2(FC) for the differentially expressed genes (under the RMA and MBEI algorithms) are shown in Figure 5.1. These putative FC values were used for the simulation. Those genes which were not found to be differentially expressed based on the regression estimates were defined to have FC=1 for the simulation.

Figure 5.1: Distributions of the putative FC values used for the RMA and MBEI algorithms.



5.2.2 Background Adjustments

RMA: The background adjustment for the RMA algorithm is detailed in Irizarry *et al.* [33] and can be easily calculated using Bioconductor. Our simulation starts with background corrected data, but in order to generate realistic data we must add some background back to the data. We began by examining the RMA background values (defined as $b_{ijkn} = PM_{ijkn} - B(PM_{ijkn})$) for a number of arrays from the barley experiment. Figure 5.2 shows background versus PM for four barley arrays. Background increases with PM until a certain point and then levels off. For the simulation, probes corresponding to the original smallest 15% of background values keep their original background value. For all other probes, a variable background term will be added by array, such that $b_{ij} \sim N(\beta, 4)$ where β is defined as the median background from the baseline array or the average of the median backgrounds from two "starter" arrays.

MBEI/dChip: According to [41], a nonspecific binding correction is performed for the MBEI algorithm. The normalization occurs before the NSB correction in the MBEI algorithm. Using dChip, we can examine the NSB values (represented as ν_{kn} from Equations 3.2 and 3.3). Based on what we have observed, this term seems to correct for optical noise rather than NSB. Table 5.1 shows the five number summaries of the background/NSB values for the two barley "starter arrays". The correlation between background and PM was 0.044 for array 1 and 0.043 for array 2. We see that these background values vary between 71 and 95 regardless of the array and do not appear to be correlated with PM. For the simulation, we use a constant background term for each array. This constant background can be estimated by the average background for a single array or the average over two arrays.

5.2.3 Estimation of Variance Components for RMA algorithm

Based on Equation 5.1 there are two variance components (σ_ϵ^2 and σ_δ^2) that need to be estimated for the RMA algorithm. Recall that ϵ_{ijkn} is an overall error

Figure 5.2: RMA background versus PM values for four barley arrays.

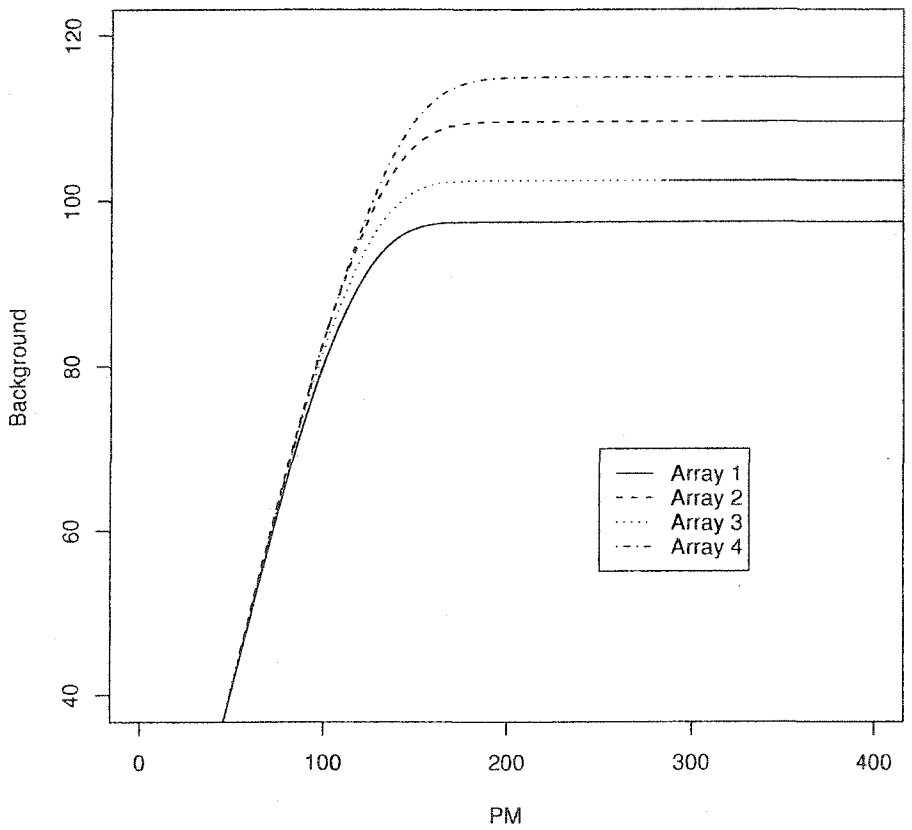


Table 5.1: Five number summaries of the dChip background adjustments for two barley arrays.

Array	Min	Q1	Median	Q3	Max
1	71	78	80	84	94
2	71	78	81	85	95

term and $\delta_{j(i)n}$ is an array within treatment error term (representing variability between biological or technical replicates). We will attempt to estimate these variance components by fitting a mixed linear model by probe set.

For the barley data, we normalized and background corrected a group of arrays using the RMA algorithm. We then imported two replicate Merit arrays and two replicate Legacy arrays into SAS. As suggested by Chu *et al.* [14], we used SAS to fit the following model:

$$logPM = Trt + Array(Trt) + Probe + (Trt \times Probe)$$

by probe set, where Trt represents cultivar (Legacy or Merit), and each cultivar was hybridized to two arrays. Array effect was considered random and all other terms were fixed. We consider only probe sets with exactly 11 probe pairs. For these probe sets the expected mean square for Array(Trt) is given by:

$$EMS(Array(Trt)) = \theta_{Array(Trt)} = \sigma^2_\epsilon + 11\sigma^2_{Array(Trt)}.$$

The values $SS_{Array(Trt)}$ and SSE were saved for each probe set.

Recall the following distributional result (with assumptions of normality and independence)

$$\frac{SS_{Array(Trt)}}{\theta_{Array(Trt)}} \sim \chi^2_{df(Array(Trt))}.$$

We can obtain a reasonable value to use for $\theta_{Array(Trt)}$ by considering a q-q plot of the $SS_{Array(Trt),n}$ values. Using ordinary least squares through the origin, we can use the slope of the regression equation as a putative value for $\theta_{Array(Trt)}$. This yields the following estimate:

$$\hat{\theta}_{Array(Trt)} = 0.4707 = \hat{\sigma}^2_\epsilon + 11\hat{\sigma}^2_{Array(Trt)},$$

so a common estimate of

$$\hat{\sigma}^2_\epsilon = \hat{\sigma}^2_{Array(Trt)} = 0.4704/12 = 0.0392.$$

Hence using the notation of Equation 5.1 we obtain $\hat{\sigma}_\delta = \hat{\sigma}_\epsilon = 0.2$.

We simulated data according for the RMA algorithm using these variance components. In order to compare the simulated data to the observed "starter" arrays we compared the plots of the PM values for the two treatments. We considered $\sigma_\delta = \sigma_\epsilon = 0.2$ and 0.15. We found that using $\sigma_\delta = \sigma_\epsilon = 0.15$ yielded simulated data that better represented the observed data, so we used this value as input for SimArray. This comparison is shown in Figure 5.3.

Previous discussion is based on the assumption that we have two replicate starter arrays per treatment for use in the simulation. In practice, it would not be realistic to assume that experimenters would have access to more than one array per treatment in order to do a sample size calculation! In this case, we would recommend using data from previously conducted similar experiments (performed with the same type of array) for estimating the variance components to be used for the sample size calculation. Subjective values based on experience may also suffice.

5.2.4 Estimation of Variance Components for MBEI algorithm

Initially for the MBEI algorithm, we attempted to use the model

$$PM_{ijkn} = N^{-1}(\nu_{kn} + \theta_{in}\Phi_{kn} + \delta_{j(i)n} + \epsilon_{ijkn}),$$

where $\delta_{j(i)n} \sim N(0, \sigma_\delta^2)$ and $\epsilon_{ijkn} \sim N(0, \sigma_\epsilon^2)$. In an attempt to estimate the variance components we imported the dChip preprocessed PM values into SAS. This time, we model the residuals from the dChip fit (defined as $resid = PM_{ijkn} - N^{-1}(\hat{\nu}_{kn} + \hat{\theta}_{in}\hat{\Phi}_{kn})$ using the model $resid = Array(Trt)$ with Array as a random effect using PROC MIXED. We used the same two replicates of each Legacy and Merit described in Section 5.2.3. We found that 83% of probe sets had Array(Trt) variance estimated to be zero. Next, we considered the standard deviation of the raw dChip residuals by probe set and array. These plots are shown in Figure 5.4. The standard deviation

Figure 5.3: (a)Observed data, (b) data simulated with $\sigma_\delta = \sigma_\epsilon = 0.2$, (c) data simulated with $\sigma_\delta = \sigma_\epsilon = 0.15$ for the RMA algorithm.

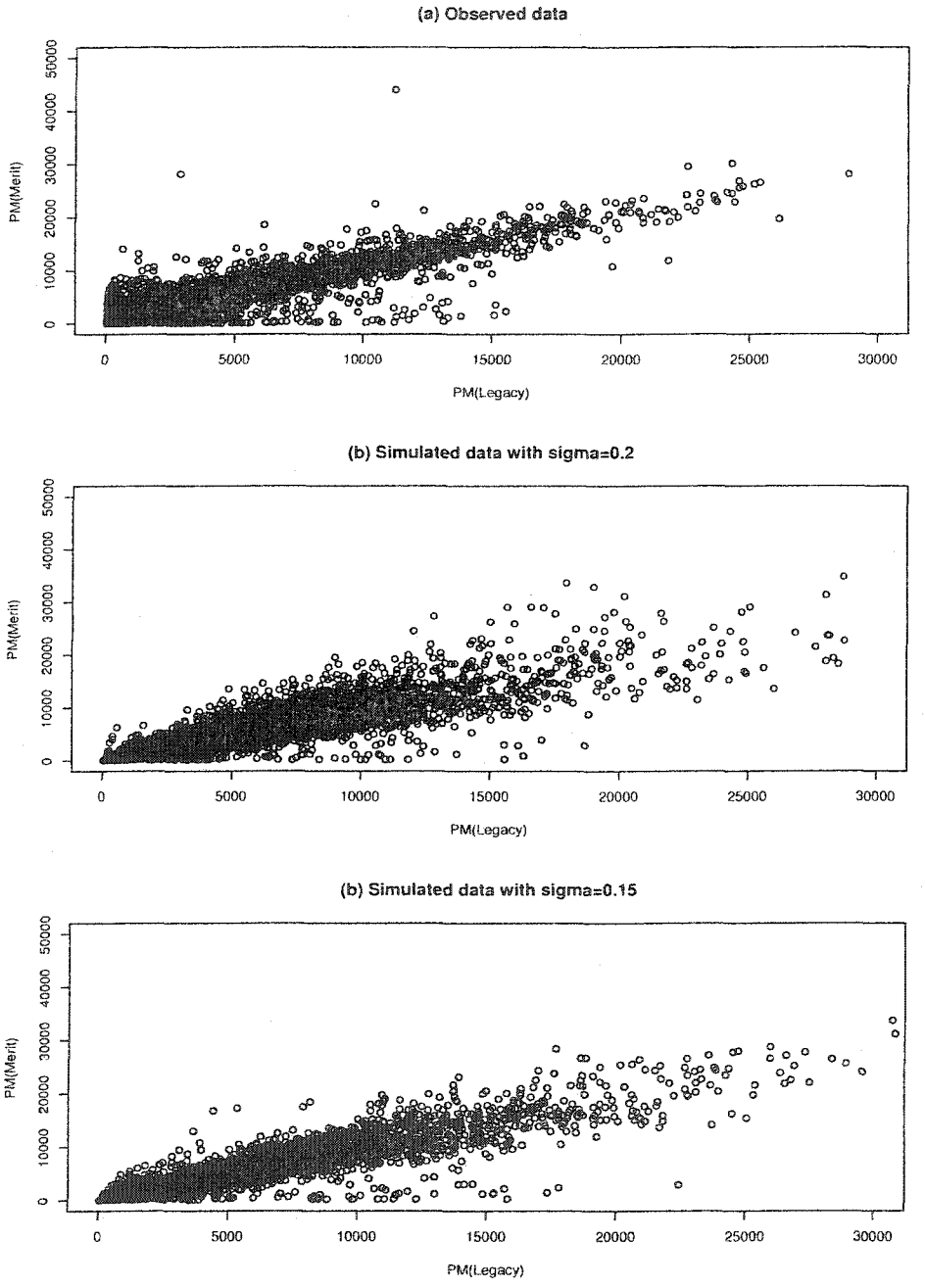
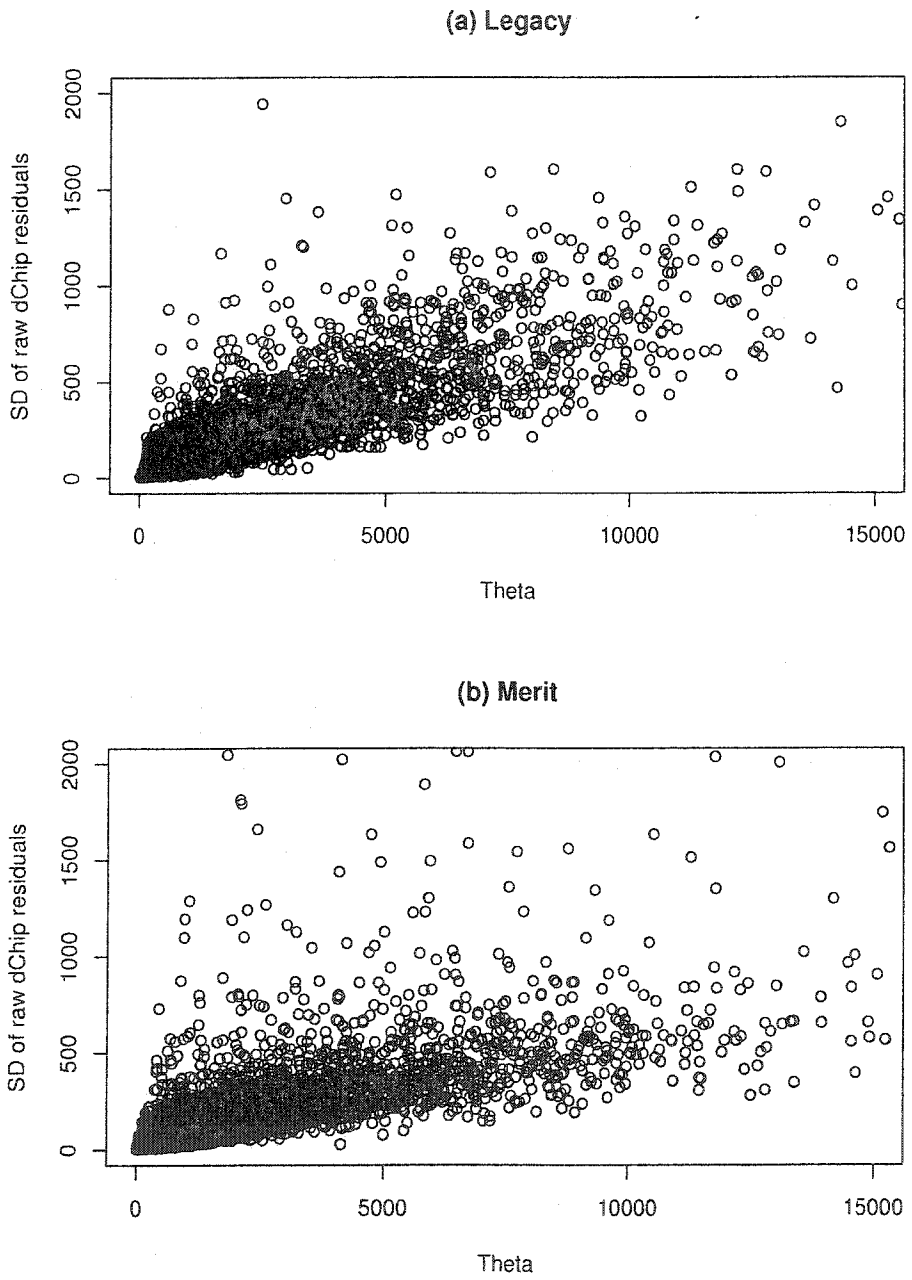


Figure 5.4: Standard deviation of raw dChip residuals versus $\hat{\theta}$ by probeset for Legacy and Merit.



tends to increase with $\hat{\theta}$. Hence, we decide that a model with two additive error terms was not necessary used Equations 5.2 and 5.3 instead.

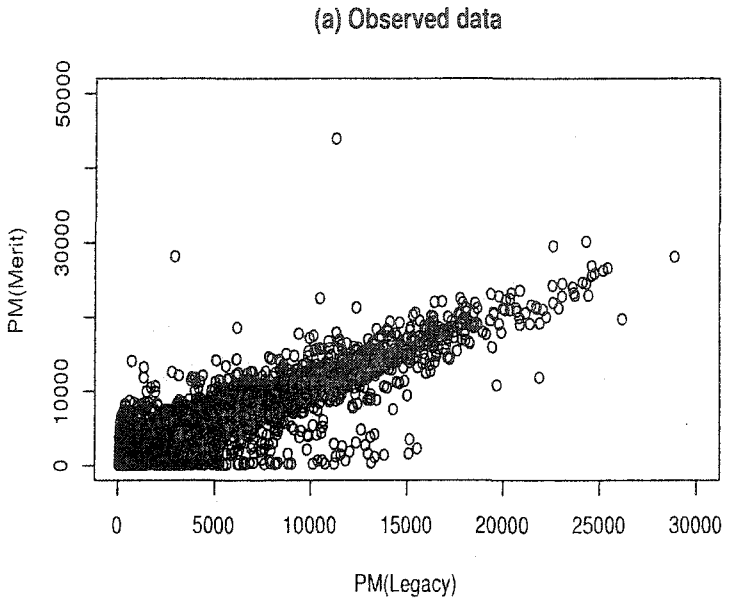
In order to estimate the value σ_ϵ^2 we used the residuals from the dChip fit and scaled them by the signal $(\hat{\theta}_{in}\hat{\Phi}_{kn})$. This implies an additive error term with standard deviation proportional to the signal. We once again used the two replicates of each Legacy and Merit. We consider the variance of these scaled residuals (by probeset and array). The median of the variances of the scaled residuals acts as an estimate of σ_ϵ^2. Taking the median value of these variances for each of the four arrays separately we found values ranging between 0.026 and 0.029. We chose to use the value $\sigma_\epsilon = 0.16$ for the sample size calculation. Comparing the plots of the PM values for the two treatments for the simulated and observed data, we found a good correspondence. This comparison is shown in Figure 5.5.

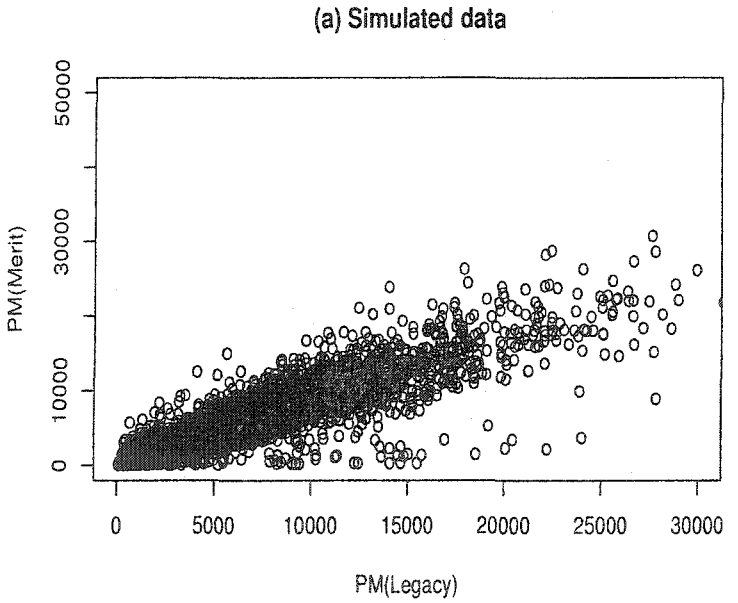
So once again, we used two replicate array per treatment to estimate the variance. However, for the MBEI analysis the single variance term (σ_ϵ^2) can be estimated with only two arrays. If an experimenter could provide a single array for each of two treatments, we could use those genes not identified as differentially expressed to estimate the variance. However, if fewer than two arrays were available, we would need to use data from other previously conducted similar experiments.

5.2.5 Analysis of Simulated Data

For MBEI we examine the results when analyzed using PM-only as well as PM-MM. In Bioconductor we used the following commands for the PM-only algorithm: expresso(data,normalize.method="invariantset",bg.correct=FALSE, pmcorrect.method="pmonly",summary.method="liwong"). This analysis does not background (or NSB) correct the data. Web documentation for the dChip PM-only algorithm states that normalization is based on the original PM and MM data, and the background correction is performed after normalization for the PM-only model

Figure 5.5: (a) Observed data and (b) data simulated (with $\sigma_\epsilon = 0.16$) for the MBEI algorithm.





(http://biosun1.harvard.edu/complab/dchip/pm%20only.htm). We do not expect this difference to have a major effect on the results. Keep in mind that Bioconductor will not return the same expression values as dChip for either the PM-only or PM-MM analysis. We chose to use Bioconductor anyway because it is much easier to use for simulation purposes.

For RMA the analysis was carried out in Bioconductor using the "rma" command.

5.3 Results for the Barley Example

All results are based on 100 runs of a given simulation scenario. All methods used a stated false discovery rate of 0.05. Results were obtained for 3,4 and 5 replicates of each treatment.

5.3.1 Detection of Differentially Expressed Genes and Error Rates

Significance testing was based on a t-test using Welch-Satterthwaite degrees of freedom. In order to adjust for multiple comparisons, we have chosen to use the Benjamini-Hochberg adjusted p-value. The Benjamini-Hochberg method (step-up false discovery rate controlling procedure) is designed to control the false discovery rate [8]. Additional details of the Benjamini-Hochberg method as well as other methods aimed at controlling the FDR are given in a paper by Reiner *et al.* [55]. Genes with adjusted p-values less than 0.05 were declared differentially expressed. The procedure is designed to control the false discovery rate at 0.05. The actual FDR will not necessarily equal 0.05. To examine this possibility we proceed as follows. For each run of the simulation, the FDR is estimated as the proportion of falsely rejected hypotheses (the proportion of genes with adjusted p-value less than 0.05 that are not differentially expressed). If no discoveries are made, then the FDR is defined to be zero.

Table 5.2: Mean and standard deviation of false discovery rate (FDR) and maximum comparison wise error rate (CWER) for the RMA and MBEI algorithms.

Method	Reps	mean(FDR)	sd(FDR)	max(CWER)
RMA	3	0.015	0.012	0.02
RMA	4	0.023	0.007	0.02
RMA	5	0.032	0.008	0.02
MBEI PM-only	3	0.019	0.009	0.02
MBEI PM-only	4	0.034	0.008	0.06
MBEI PM-only	5	0.048	0.012	0.18
MBEI PM-MM	3	0.014	0.012	0.02
MBEI PM-MM	4	0.023	0.008	0.02
MBEI PM-MM	5	0.028	0.007	0.03

We also evaluated the false positive rates, on a gene by gene basis (still after an adjustment for multiple comparisons). We call this the comparison wise error rate (CWER). An estimate of CWER is the proportion of times the adjusted p-value for an "unchanged" gene was less than 0.05 out of the 100 simulation replications. Each gene has its own type 1 error rate.

The mean and standard deviation of the FDR as well as the maximum CWER for each of the scenarios (for both RMA and MBEI) are given in Table 5.2. These results are discussed in Section 5.4.

5.3.2 Power

In order to estimate the power for a given scenario, we calculated, for each gene, the proportion of the 100 simulation replications where the adjusted p-value was less than 0.05. We expect this proportion to be small when a gene is not differentially expressed and large when the fold changes are large in either direction. After using a given algorithm to generate and analyze the data for each run of the simulation, we first plotted the estimated power as a function of $|log_2(FC)|$ for the barley1 genes. A loess curve was fitted in order to smooth the plot. The fitted power curves for the the RMA algorithm (with 3,4 and 5 reps) are shown in Figure 5.6a. The fitted

power curves for the the MBEI algorithm (with 3,4 and 5 reps) are shown in Figures 5.6b and c.

In addition to power, some researchers are interested in the sample size required to detect a certain number of the most highly differentially expressed genes. For example a researcher may want to know what proportion of the true top M most highly differentially expressed genes are captured in the observed top N genes. Based on a list of the top N most highly differentially expressed genes in the experiment, they could verify expressions for some or all of these genes using real-time PCR or a northern blot. Clearly these values of M and N will vary based on the investigator. The idea is similar to the study by Rosati *et al.* [58], where they identified small groups of genes from a microarray experiment and verified their results using real-time PCR. Here we consider what proportion of the true top 50 most highly differentially expressed genes are captured in the observed top 100 most highly differentially expressed genes based on simulated data. This proportion can be calculated for each run of the simulation. We show box plots of these proportions in Figure 5.7 for both RMA and MBEI. This information is summarized in tabular form in Table 5.3.

Table 5.3: Five number summary of the proportion of the true 50 most highly expressed genes captured in the observed 100 most highly expressed genes by method and number of replicates.

Method	Reps	Min	Q1	Median	Q3	Max
RMA	3	0.64	0.74	0.78	0.82	0.88
RMA	4	0.88	0.90	0.90	0.90	0.92
RMA	5	0.88	0.90	0.90	0.90	0.92
MBEI PM-only	3	0.66	0.80	0.84	0.86	0.94
MBEI PM-only	4	0.94	0.96	0.96	0.98	1.00
MBEI PM-only	5	0.94	0.96	0.96	0.98	1.00
MBEI PM-MM	3	0.6	0.74	0.8	0.84	0.96
MBEI PM-MM	4	0.92	0.96	0.96	0.98	1
MBEI PM-MM	5	0.94	0.96	0.98	0.98	1

Figure 5.6: Power curves for RMA and MBEI.

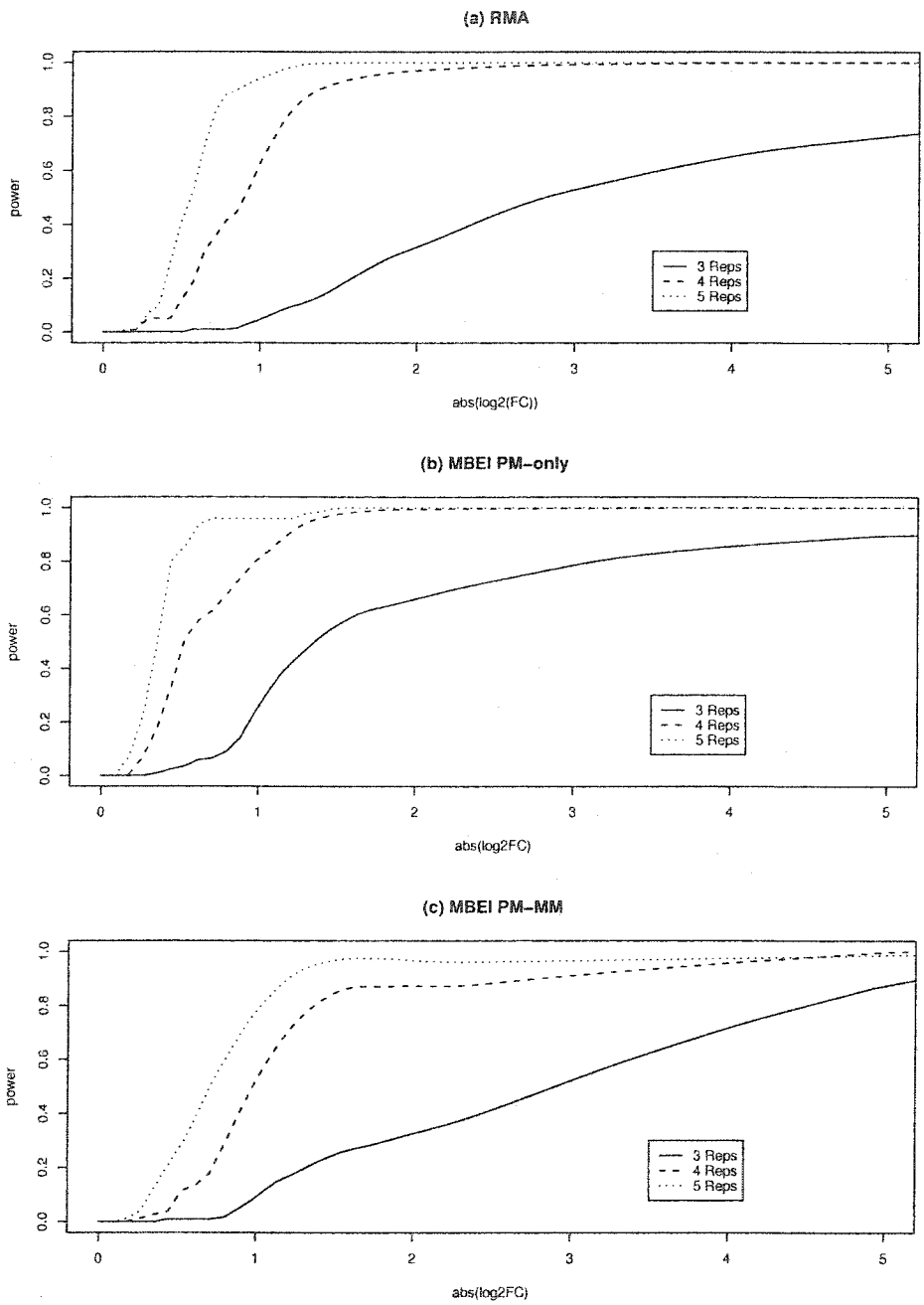
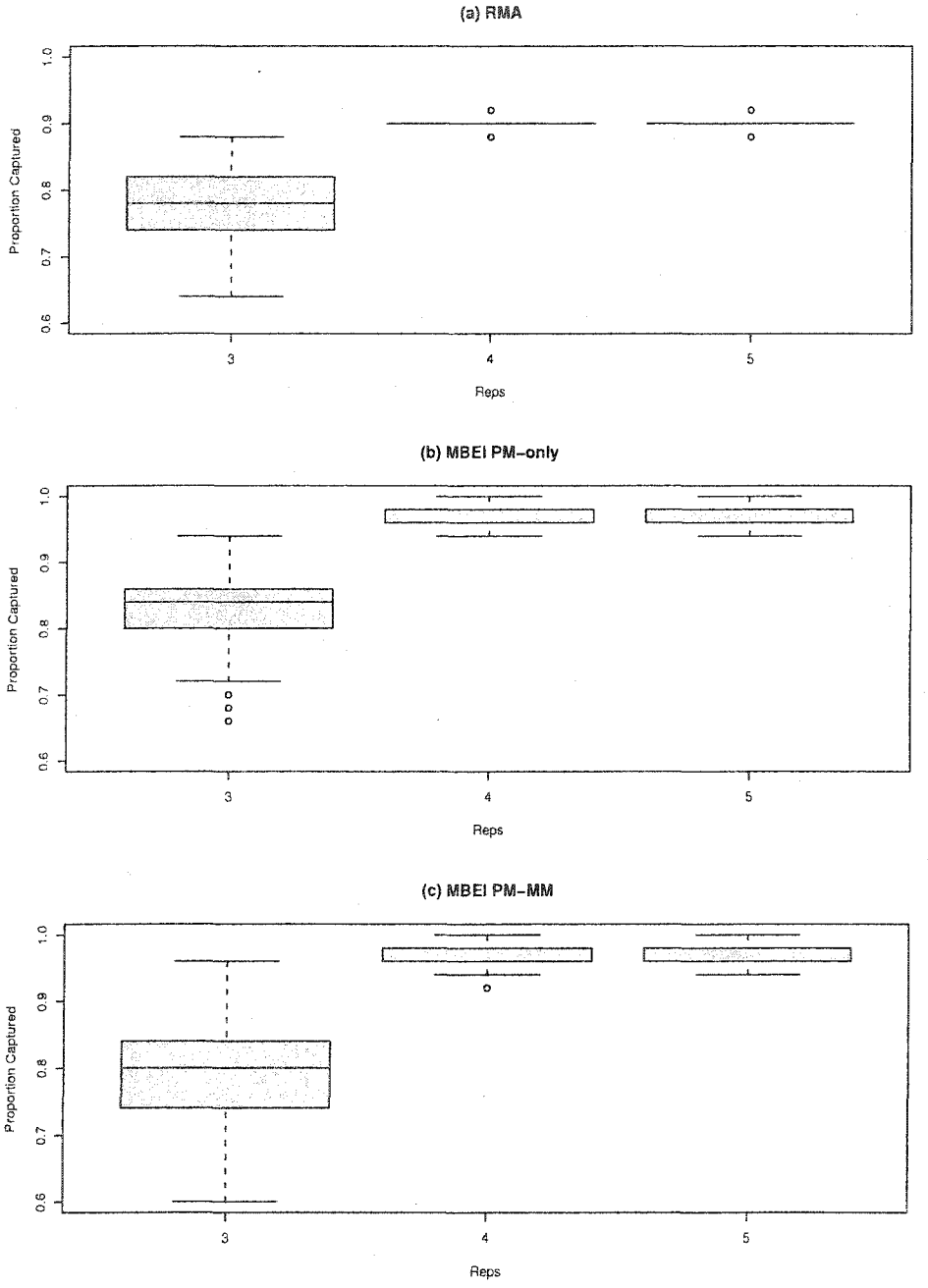


Figure 5.7: Box plots of the proportion of the true 50 most highly expressed genes captured in the observed 100 most highly expressed genes by method and number of replicates.



5.4 Discussion of Results

It is important to note that the results for RMA and MBEI cannot be directly compared because they employ different algorithms and error structures.

When considering the FDR, we note that both RMA and MBEI show increasing mean(FDR) with increasing replicates. However both methods remain below 0.05. The maximum type 1 error rate stays approximately constant for RMA and MBEI PM-MM, but increases with replicates for MBEI PM-only. Note that maximum CWER reaches a high of 0.18 for MBEI PM-only with five replicates.

The power plots (Figure 5.6) for both methods show that substantial gains are made when the number of replicates is increased from three to four. However, only modest gains are achieved when the number of replicates is increased from four to five. Both RMA and MBEI have the lowest power with only three replicates. However, the power for RMA and MBEI PM-MM is rather low in this situation. The estimated power for RMA and MBEI PM-MM with three replicates is less than 0.4 with a fold change of ± 4. Note that MBEI PM-only method reaches an estimated power of about 0.6 under the same conditions.

The box plots summarizing the proportion of the true top 50 most highly expressed genes captured in the observed top 100 genes (Figure 5.7) for each method show once again that substantial gains are made when the number of replicates is increased from three to four. We see almost no difference when the number of replicates is increased from four to five.

In conclusion, we feel that SimArray offers a very flexible and powerful tool for the determination of sample size for microarray experiments.

Chapter 6

NORMALIZATION ISSUES AND COMPARISON OF METHODS

Normalization attempts to correct for systematic array differences in order to better detect differentially expressed genes across treatments. These array differences stem from possible differences in the relationship between total signal and observed intensity. This relationship can be described using a calibration function. In practice since each sample is randomly assigned to an array, if we had many replicates of each treatment these array differences would "average out". However, since the number of replicates per treatment is usually small (less than 10) for microarray experiments, normalization is performed in order to increase power. So, by reducing within treatment differences, we will be able to identify more differentially expressed genes.

Now we consider what other authors have said about normalization. Hoffmann *et al.* [30] describe normalization as "pre-scaling of the fluorescence intensity across different arrays belonging to one experiment to correct for differences in probe labelling, probe concentration, hybridization efficiency, and potentially other factors". Schadt *et al.* [59] say "Normalizing multiple probe arrays to allow direct array-to-array comparisons presents one of the greatest challenges in expression array data analysis". Bolstad *et al.* [11] say "...observed expression levels also include variation that is introduced during the process of carrying out the experiment, which could be classified as obscuring variation. Examples of this obscuring variation arise due to differences in sample preparation (for instance labelling differences), production

of the arrays and the processing of the arrays (for instance scanner differences). The purpose of normalization is to deal with this obscuring variation".

In this chapter, we examine the effect of the calibration function on estimability of total signal ratio. We define total signal ratio (TSR) as the ratio of the total signal (including both GSB and NSB) across arrays or treatments. We present some commonly used approaches to normalization. For quantile normalization (associated with RMA) and invariant set normalization (associated with dChip), we present some small examples illustrating their performance. We also examine the results of a simulation experiment comparing the performance of quantile, invariant set and global normalization methods.

6.1 Impact of the Calibration Function on Estimability of Signal Ratios

The relationship between total signal (including both GSB and NSB) and observed intensity may not be the same from array to array. Suppose that S_{1k} is the total signal for probe k and sample 1 and S_{2k} is the total signal for sample 2. Let f_1 be the function relating total signal to the observed fluorescence intensity for array 1 and f_2 be the function relating total signal to intensity for array 2. In the absence of random errors, $I_{1k} = f_1(S_{1k})$ would be the observed value of probe k for sample 1 on array 1 and $I_{2k} = f_2(S_{2k})$ would be the observed value of sample 2 on array 2. Note that in practice the probe level total signal values (S_{ik}), f_1 and f_2 are all unknown and only the intensity values (I_{ik}) are observed. Clearly the total signal ratio (TSR=$S_{ik}/S_{i'k}$) is well-defined but unknown. The question is whether or not this ratio is estimable.

Note that for this discussion we consider the probe level total signal, where total signal is the sum of gene specific binding (GSB) and nonspecific binding (NSB). If we assume that nonspecific binding (ν) is a function only of probe sequence, then the NSB for a given probe will fixed regardless of treatment or array. Hence (in the

absence of random error) intensity is related to signal as

$$I_{ikn} = f_i(\Phi_{kn}\theta_{in} + \nu_{ikn}) = f_i(GSB_{ikn} + NSB_{kn}) = f_i(S_{ikn}).$$

Our interest lies in examining the impact of the calibration function on the estimation of the TSR. Estimation of or correction for nonspecific binding is discussed elsewhere in this dissertation.

We now consider a number of possible forms for the calibration function. In the following discussion, we consider probe level values. For simplicity, we drop the probe set subscript (n).

Scale calibration function with no random error: Let us consider the case where intensity is directly proportional to total signal. Then $I_{1k} = f_1(S_{1k}) = f_1 \times S_{1k}$ and $I_{2k} = f_2(S_{2k}) = f_2 \times S_{2k}$. If we consider the ratios I_{2k}/I_{1k}, we would find that for unchanged probes (with $S_{1k} = S_{2k}$) the ratio would be exactly equal to f_2/f_1. If we assume that the majority of probe sets are unchanged, then we can use the mode of the empirical distribution of probe level intensity ratios (I_{2k}/I_{1k}) to estimate f_2/f_1. Note that since we are assuming no error, the ratio f_2/f_1 is *known*. Hence, we can find the TSR as

$$\frac{S_{2k}}{S_{1k}} = \frac{f_1 \times I_{2k}}{f_2 \times I_{1k}}.$$

So, the TSR is estimable in this case.

Linear calibration function with no random error: Now let us consider the case where intensity is not proportional to the signal but is linearly related to signal. Then $I_{1k} = f_1(S_{1k}) = a_1 + b_1 S_{1k}$ and $I_{2k} = f_2(S_{2k}) = a_2 + b_2 S_{2k}$. The value a can be thought of as background intensity associated with the scanning process.

In an attempt to normalize array 2 to array 1, one approach is to fit a line to the pairs of unchanged probes. Unchanged probes (for which $S_{1k} = S_{2k}$) would fall directly on a line. Hence in order to identify the group of unchanged probes,

we would identify the largest group of probes with correlation=1. Fitting a line to these unchanged probes we would find that

$$I^*_{2k} = f_1(f_2^{-1}(I_{2k})) = (a_1 - a_2 b_1/b_2) + b_1/b_2 I_{2k} = a^* + b^* I_{2k},$$

where I^*_{2k} indicates the normalized value for probe k and sample 2. This normalization relates the observed intensities from array 2 (I_{2k}) to the observed intensities from array 1 (I_{1k}).

Many authors suggest normalizing one array to another and then using the ratio of the normalized intensities as an estimate of the fold change. If we tried to find the TSR using I_{1k} and I^*_{2k}, we would consider

$$I^*_{2k}/I_{1k} = f_1(S_{2k})/f_1(S_{1k}).$$

For an unchanged probe (where $S_{1k} = S_{2k}$) we would find

$$I^*_{2k}/I_{1k} = f_1(S_{2k})/f_1(S_{1k}) = 1$$

as expected. However, for a changed probe (where $S_{2k} = cS_{1k}$), we would find

$$I^*_{2k}/I_{1k} = f_1(cS_{1k})/f_1(S_{1k}) \neq c.$$

Note that the estimated TSR is dependent on which array is normalized because

$$I^*_{2k}/I_{1k} = f_1(cS_{1k})/f_1(S_{1k}) \neq f_2(cS_{1k})/f_2(S_{1k}) = I_{2k}/I^*_{1k}.$$

This issue is illustrated in Example 1.

If the data is assumed to be background corrected (using independent information), then we have reduced the problem to that of a estimating the TSR with a scale calibration function.

Example 1: Let us look at a very small array with only 10 observations. The data is shown in Table 6.1. The known (total) signal for probe k is represented

Table 6.1: Data for Example 1.

S_1	S_2	I_1	I_2	I_1^*	I_2^*	TSR	I_2^*/I_1	I_2/I_1^*
1	1	9	6	6	9	1	1.00	1.00
2	2	14	10	10	14	1	1.00	1.00
3	6	19	26	14	34	2	1.79	1.86
3	3	19	14	14	19	1	1.00	1.00
5	5	29	22	22	29	1	1.00	1.00
6	2	34	10	26	14	0.33	0.41	0.38
7	7	39	30	30	39	1	1.00	1.00
8	4	44	18	34	24	0.5	0.55	0.53
8	8	44	34	34	44	1	1.00	1.00
9	9	49	38	38	49	1	1.00	1.00

as S_{1k} for the control sample and S_{2k} for the treatment sample. There is a linear relationship between total signal and observed intensity on each of the arrays. For array 1, $I_{1k} = f_1(S_{1k}) = 4 + 5S_{1k}$ and for array 2, $I_{2k} = f_2(S_{2k}) = 2 + 4S_{2k}$. So, $f_1^{-1}(x) = -0.8 + 0.2x$ and $f_2^{-1}(x) = -0.5 + 0.25x$. In order to normalize array 2 to array 1, we would use $f_1(f_2^{-1}(x)) = 1.5 + 1.25x$. To normalize array 1 to array 2, we would use $f_2(f_1^{-1}(x)) = -1.2 + 0.8x$. We see that in either case, the estimated TSR is always correct for unchanged probes but not for changed probes. Furthermore, the two normalizations yield different estimated ratios.

Nonlinear calibration with no random error: Clearly if a nonlinear calibration function is considered, the same estimability issues will be present as in the linear case.

6.2 Normalization Methods

We now present some commonly used normalization methods.

6.2.1 Scale Normalization

Recall that MAS computes scaling (sf) and normalization factors (nf) and the reported value for a probe set is:

$$ReportedValue(i) = nf * sf * 2^{SignalLogValue_i}. \quad (6.1)$$

The Affymetrix Statistical Algorithms Reference Guide [2] states "Normalization and scaling techniques can be applied by using data from a selected user-defined group of probe sets, or from all probe sets. When normalization is applied the intensity of the probe sets (or selected probe sets) from the experimental array are normalized to the intensity of the probe sets (or selected probe sets) on the baseline array. When scaling is applied, the intensity of the probes sets (or selected probe sets) from the experimental array and the intensity of the probe sets (or selected probe sets) from the baseline array are scaled to a user-defined target intensity." They also (vaguely) describe a " 'robust normalization,' which is not user modifiable, accounts for unique probe set characteristics due to sequence dependent factor such as affinity of the target to the probe and linearity of hybridization of each probe pair in the probe set" [2].

There has been some criticism of the scale normalization method used by Affymetrix. Bolstad *et al.* [11] say "Affymetrix has approached the normalization problem by proposing that intensities should be scaled so that each array has the same average value. The Affymetrix normalization is performed on the expression summary value. This approach does not deal particularly well with cases where there are nonlinear relationships between arrays."

6.2.2 Quantile Normalization

Quantile normalization was developed by Bolstad *et al.* [11]. This method forces the distribution of probe intensities for each array (in a group of arrays) to be the same. They employ the following algorithm:

1. Given n microarrays with p probes, form a matrix X of dimension $p \times n$ where data from each array are placed in one of the columns.

2. Sort each column of X to give X_{sort}.

3. Take the means across rows of X_{sort} and assign this mean to each element in the row to get X'_{sort}.

4. Get $X_{normalized}$ by unsorting each column of X'_{sort} to have the same ordering as original X.

The authors note that forcing the quantiles to be exactly equal may be a problem. Clearly, each array will have the same set of probe values (although assigned to different probes). Note that for this method, the arrays are not normalized to a baseline array, but are instead simultaneously normalized to a conceptual average (baseline) array.

Example 2: Once again we will restrict ourselves to small example which illustrates the issues. The known (total) signal value for probe k is represented as S_{1k} for the control sample and S_{2k} for the treatment sample. Note that the values for the treatment array are just a permutation of the control array values. In other words we assume that the array values are identically distributed. In this case, there is a scale relationship between total signal and observed intensity on each of the arrays. For array 1, $I_{1k} = f_1(S_{1k}) = S_{1k}$ and for array 2, $I_{2k} = f_2(S_{2k}) = 2S_{2k}$. Here we consider the effect of quantile normalization on the estimation of the TSR.

Table 6.2: Data for Example 2 where we apply quantile normalization to identically distributed arrays with scale calibration functions.

$S_1 = I_1$	S_2	I_2	I_1^*	I_2^*	TSR	I_2^*/I_1^*
1	4	8	1.5	6	4	4
2	2	4	3	3	1	1
3	3	6	4.5	4.5	1	1
4	1	2	6	1.5	0.25	0.25
5	5	10	7.5	7.5	1	1

So in this example where the array values are identically distributed and with scale calibration functions, quantile normalization performs correctly. Note that if these assumptions hold, then the normalization function (applied to the observed intensity data) can be expressed as:

$$I_{ik}^* = N(I_{ik}) = 0.5(s_1 + s_2)S_{ik} = s^* S_{ik},$$

where s_1 is the scale calibration factor for array 1 and s_2 is the scale calibration factor for array 2. Hence,

$$\frac{I_{ik}^*}{I_{i'k}^*} = \frac{s^* S_{ik}}{s^* S_{i'k}} = \frac{S_{ik}}{S_{i'k}}.$$

So the estimated TSR will be identical to the true TSR for all probes.

Example 3: Here we again start with identically distributed array values, but allow a linear calibration function. For array 1, $I_{1k} = f_1(S_{1k}) = S_{1k}$ and for array 2, $I_{2k} = f_2(S_{2k}) = 1 + 0.5S_{2k}$.

So in this example where the array values are identically distributed and the calibration functions are linear, quantile normalization achieves the correct TSR for non-differentially expressed genes but not for the differentially expressed genes. The reason is clear when we consider the normalization function:

$$I_{ik}^* = N(I_{ik}) = 0.5(f_1 + f_2)S_{ik} = a^* + b^* S_{ik}.$$

Table 6.3: Data for Example 3 where we apply quantile normalization to identically distributed arrays with linear calibration functions.

$I_1 = S_1$	S_2	I_2	I_1^*	I_2^*	TSR	I_2^*/I_1^*
1	4	3	1.25	3.5	4	2.8
2	2	2	2	2	1	1
3	3	2.5	2.75	2.75	1	1
4	1	1.5	3.5	1.25	0.25	0.36
5	5	3.5	4.25	4.25	1	1

(Note that if $f_1(x) = a_1 + b_1 x$ and $f_2(x) = a_2 + b_2 x$ then $a^* = 0.5(a_1 + a_2)$ and $b^* = 0.5(b_1 + b_2)$.) Hence,

$$\frac{I_{ik}^*}{I_{i'k}^*} = \frac{a^* + b^* S_{ik}}{a^* + b^* S_{i'k}}.$$

So, the correct TSR is preserved for non-differentially expressed probes, but not for those probes which are differentially expressed.

Example 4: Now we consider an example where the array 2 values are not simply a permutation of the array 1 values. We assume the same scale calibration functions as in Example 2; $I_{1k} = f_1(S_{1k}) = S_{1k}$ and $I_{2k} = f_2(S_{2k}) = 2S_{2k}$.

Table 6.4: Data for Example 4 where we apply quantile normalization non-identically distributed arrays with scale calibration functions.

$I_1 = S_1$	S_2	I_2	I_1^*	I_2^*	S_2/S_1	I_2^*/I_1^*
1	6	12	1	8.5	6	8.5
2	2	4	3	3	1	1
3	3	6	4.5	4.5	1	1
4	0.5	1	7	1	$\frac{1}{8}$	$\frac{1}{7}$
5	5	10	8.5	7	1	0.824

Here the estimated TSRs for both differentially expressed and non-differentially expressed probes are affected by the normalization algorithm.

So, quantile normalization performs well when only a scale calibration function is creating array to array differences and the probe values are identically distributed

for two or more arrays. If these conditions do not hold, quantile normalization cannot correctly recover the correct TSR values. Note that in each of the quantile normalization examples presented, we only consider the effect of the normalization and ignore (do not perform) any background correction. However, using the RMA background correction will not resolve the issues revealed here because the background correction is performed "by probe" with a different background estimated for each probe.

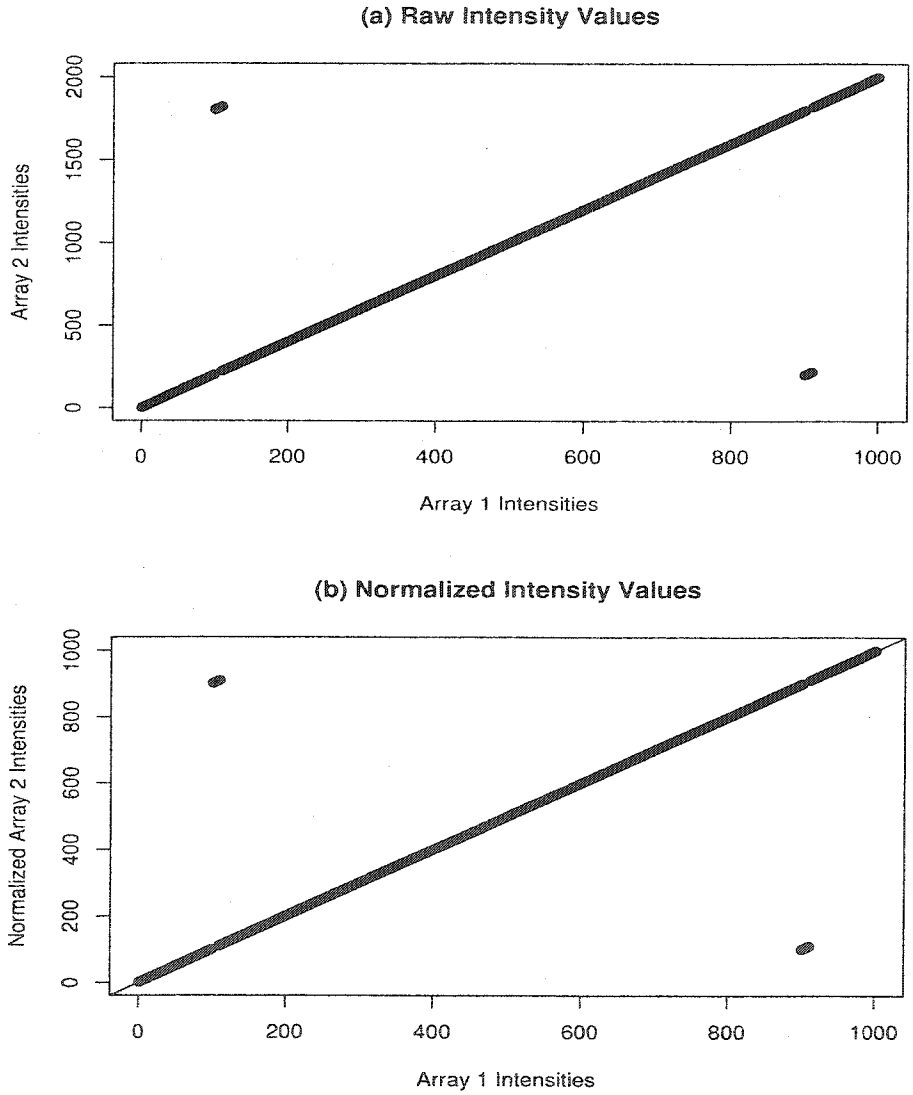
6.2.3 Invariant Set Normalization (IVSN)

Li and Wong ([59],[42]) propose an invariant set normalization method. The first step of the process is to find a set of unchanged probes (on which to base the normalization). They search for a set of invariant probes based on rank. If a probe's ranks are the same (or close to the same) in the baseline and experimental arrays, then this probe will be added to the invariant set. They provide an iterative algorithm that chooses a large set of probes as belonging to the invariant set. After invariant set is selected, a piecewise linear running median line is calculated (for the invariant set) and used as the normalization function. Note that for this method, all arrays are normalized to one baseline array.

Example 5: In order to examine dChip performed invariant set normalization, we need to work with a larger data set. Here we consider 1000 probe values for two arrays. The known (total) signal value for probe k is represented as S_{1k} for the control sample and S_{2k} for the treatment sample. We consider a scale calibration function relating intensity to total signal. For array 1, $I_{1k} = f_1(S_{1k}) = S_{1k} = k$ and for array 2, $I_{2k} = f_2(S_{2k}) = 2S_{2k}$. A total of 20 probes were differentially expressed: $S_{2,101} = 901...S_{2,110} = 910$ (TSR ≈ 9) and $S_{2,901} = 101...S_{2,910} = 110$ (TSR $\approx \frac{1}{9}$). This data was normalized using dChip. The raw and normalized intensity values are shown in Figure 6.1. dChip successfully recovered the correct TSR values for all 20

differentially expressed probes and most of the unchanged probes. For the smallest and largest 20 probes, an artifact was introduced after normalization. Specifically, for k=1...20 and 981...1000, $I^*_{2k} = I_{1k} + 0.5$ even though $S_{1k} = S_{2k}$ for these probes. This yielded estimated TSR values slightly off from the true TSR=1 for these probes.

Figure 6.1: Raw and dChip normalized intensity values for Example 5.



Example 6: We again consider 1000 probe values. We consider a linear calibration function relating intensity to total signal. For array 1, $S_{1k} = k$ and $I_{1k} = f_1(S_{1k}) = 5 + 0.9S_{1k}$ and for array 2, $I_{2k} = f_2(S_{2k}) = 10 + 1.5S_{2k}$. A total of 20 probes were differentially expressed: $S_{2,101} = 901...S_{2,110} = 910$ (TSR ≈ 9) and $S_{2,901} = 101...S_{2,910} = 110$ (TSR $\approx \frac{1}{9}$). This data was normalized using dChip. The raw and normalized intensity values are shown in Figure 6.2. In this example, dChip underestimated the up-regulated TSR values (probes 101-110) and overestimated the down-regulated TSR values (probes 901-910). dChip successfully recovered the TSR values for most of the unchanged probes. However, the same artifact seen in Example 5 (affecting the same probes) was seen again here.

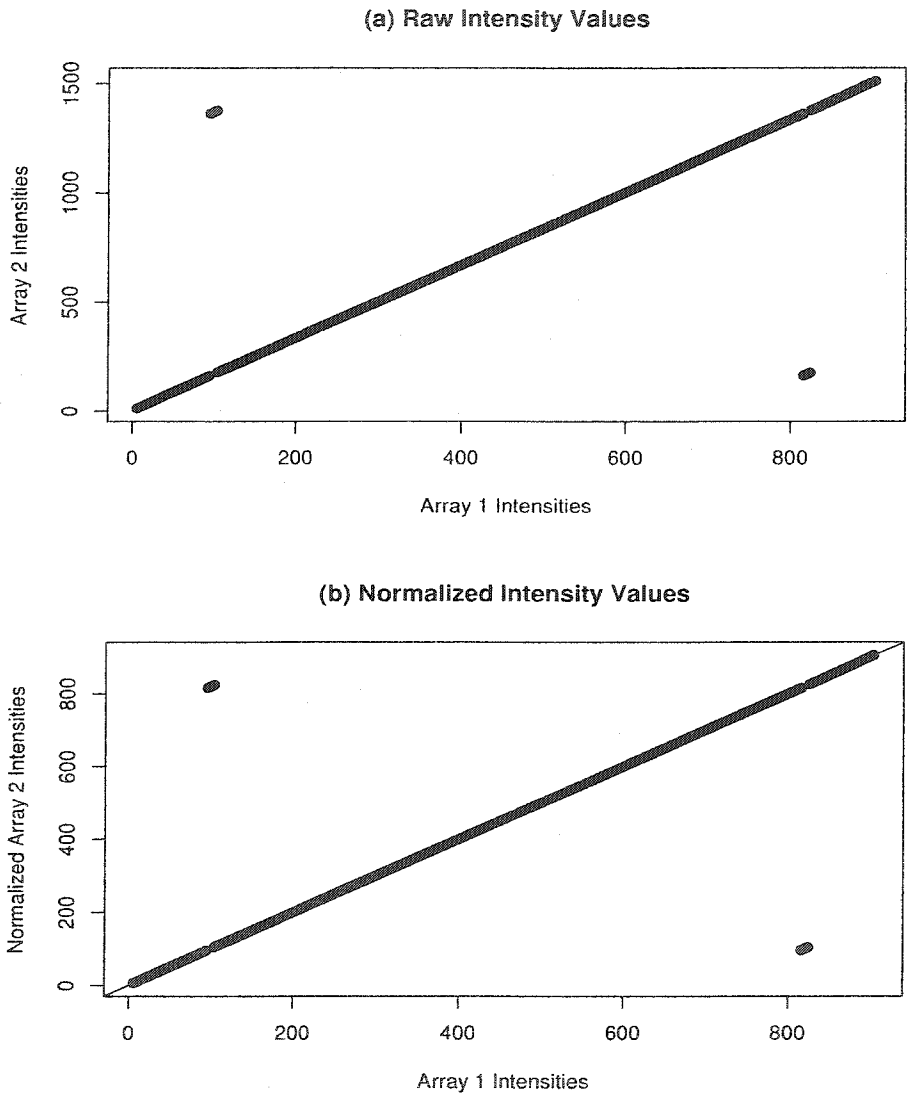
So, besides a small artifact introduced during the normalization process, invariant set normalization performs well when only a scale calibration function is creating array to array differences. If even a linear calibration function is at work, then invariant set normalization cannot recover the correct TSR values for differentially expressed probes.

6.2.4 More Normalization Methods

Qspline normalization was proposed by Workman *et al.* [75]. The goal is similar to that for quantile normalization. Their approach "seeks to transform the distribution of one array to the distribution of a target array" [75]. In order to accomplish this, qspline normalization uses quantiles from a target and baseline array to fit smoothing splines. The splines are used as a normalization function.

Hill *et al.* [28] introduce a normalization method based on spike-in values. So, instead of basing the normalization on all probes (as is done for a scale normalization or quantile normalization) or unchanged probes (as is done in invariant set normalization), the normalization is based on a small number of spike-in probe sets present on all Affymetrix GeneChip arrays. A possible benefit of this method is that a user

Figure 6.2: Raw and dChip normalized intensity values for Example 6.



might be able to estimate the calibration function, instead of just normalizing one array to another. A possible problem with this method is that the normalization would be based on a small number of probe sets. The authors also propose a cross between the spike-in and scale normalization methods.

Kepler *et al.* [35] discuss a method of normalization based on self-consistency. This iterative process is based on the assumption most genes are unchanged. So, starting with all probe sets as "core genes", they normalized all genes against the "core genes", then define the new core as those genes which appear to be unchanged after normalization. They iterate until they reach a convergence set of "core genes". Normalization (based on this group of "core genes") is performed using local regression. This normalization allows for nonlinear function.

Zien *et al.* [80] propose a two-step method for scale calibration. First they propose estimating a "quotient of the constants of proportionality" (q^*_{ij}) between each pair of arrays. For arrays i and j they consider the empirical distribution of the ratios of background corrected (probe) measurements (for the "set of genes that are considered to be expressed and reliably measured"). Using the empirical distribution of these observed ratios, they propose a number of different ways to estimate q^*_{ij} (for example the mean or the median of the observed ratios). Then using each of the $\hat{q}^*_{ij}$ values and taking a maximum likelihood approach, they determine a set of scaling factors for the expression indices. The authors say "given such (scaling) values, [they] can make the measured expression level value $l_{g,k}$ mutually comparable between different samples k by rescaling them". So, they find initial scale values using probe level data, but then rescale the expression values.

6.3 Comparison of normalization methods by simulation

Normalization methods (and their effect on subsequent analyses) have been compared in a number of papers. Hoffmann *et al.* compare results when using

four different normalization techniques and three different algorithms for identifying differentially expressed probe sets [30]. Two methods based on invariant-feature normalization (dChip) [42], invariant-set normalization ("The Equalizer")[66], and global scaling were considered. For the detection of differentially expressed probe sets, SAM (significance analysis of microarrays) and both the F-test for parametric ANOVA and the H (Kruskal-Wallis) test for non-parametric ANOVA were considered. Hoffmann *et al.* concluded that "normalization has a profound influence on detection of differentially expressed genes. This influence is higher than that of three subsequent statistical analysis procedures examined" [30]. Bolstad *et al.* [11] compare the performance of five normalization methods (including quantile, scale and invariant set) using spike-in and dilution data. They find that the quantile method "performed favorably, both in terms of speed and when using our variance and bias criteria, and therefore should be used in preference to other methods." As part of a larger comparison, Choe *et al.* [13] found that there was no clear preference between normalization methods.

In order to compare the performance of each of the methods, a statistical simulation study was conducted in R using BioConductor. The following model will be used in the simulation (for treatment i, array j, probe k, probe set n):

$$PM_{ijkn} = b_{ij} + f_{ij}(\Phi_{kn}\theta_{in})$$

$$MM_{ijkn} = b_{ij} + f_{ij}(\phi_{kn}\theta_{in})$$

where b is the background and f is the calibration function for a given array. This is the same simulation model introduced for the comparison of methods using SimArray (Equations 4.1 and 4.2), with no random error.

Before starting the simulation, we choose a microarray study for which raw data are available and select one array from this study. This array will be used as the "template" for the simulation study. Subtract an estimated background intensity

$(\hat{b_{ij}})$ defined as the minimum intensity of all probes. If we assume $f_{ij}(x) = 1$ and no error or further background, then we are left with the the signal values.

For each run of the simulation create the "true" experimental array by multiplying all (PM and MM) probes for a given probe set by an appropriate fold change value. So, all probe pairs of the same probe set are defined to have the same signal fold change. We randomly assigned 2% of all probe sets to represent differentially expressed genes. For these differentially expressed genes, the fold changes are based on a gamma distribution. Recall that the gamma distribution allows for values greater than zero. For a given realization from the gamma distribution (x), the signal fold change was defined as either x+1 (for an up-regulated gene) or 1/(x+1) (for a down-regulated gene).

Now that we have a true baseline and experimental array, we create the observed arrays using the simulation model with zero background:

$$PM_{ijkn} = f_{ij}(\Phi_{kn}\theta_{in})$$

$$MM_{ijkn} = f_{ij}(\phi_{kn}\theta_{in}).$$

Hence we apply only a calibration function (f). The calibration function will allow for scale and nonlinear response patterns relating observed intensity to signal.

For a given run of the simulation the set of arrays is normalized using quantile, qspline and scale normalization. Note that the scale normalization is not MAS scale normalization, because we are normalizing at the probe level while MAS scale normalization is applied at the expression or probe set level.

We simulate only a single baseline and experimental array. For the differentially expressed genes, the signal fold changes were generated using a gamma distribution with a shape parameter of either $\lambda = 2$ or 4 and with a scale parameter of one. We consider two calibration functions - the identity function and a nonlinear calibration function.

For this simulation study all runs for all scenarios start with data from a single array. The array data was taken from a spike-in experiment where 11 different cRNA fragments were added to the hybridization mixture of HGU95A GeneChip arrays at varying concentrations [28]. This data is available from GeneLogic at http://qolotus02.genelogic.com/datasets.nsf/. Array 92456hgu95a11 was used here. There are a total of 12,626 probe sets for this type of array.

6.4 Simulation Results and Discussion

The following quantity is considered for unchanged probes only: $|log_2(I^*_{e.kn}/I^*_{b.kn})|$. This represents the log of the normalized intensity ratio. For unchanged probe sets, $S_{b.kn} = S_{e.kn}$, so ideally $|log_2(I^*_{e.kn}/I^*_{b.kn})| = 0$. If there is a 1% error or less this "logratio" will be less than 0.0145, and if there is a 2% error or less the logratio will be less than 0.029. For constant normalization, the logratio is recorded for each run (because every probe has the same value). For the other normalization methods, the proportion of probes meeting the 1% and 2% criteria are recorded.

The empirical cdf functions for the scenarios with $\lambda = 2$ and $\lambda = 4$ with identity calibration function (and no error) are shown in Figure 6.3. These results are based on 500 runs of the simulation. With $\lambda = 2$, identity calibration and no error, 12.8% of runs had a 1% error or less and 95.4% of runs had a 2% error or less for scale normalization. With $\lambda = 4$, identity calibration and no error, only 0.4% of runs had a 2% error or less for scale normalization.

The empirical cdf functions for the scenarios with $\lambda = 2$ and nonlinear calibration function and no error are shown in Figure 6.4. These results are based on 100 runs of the simulation. For scale normalization, 13% of runs had a 1% error or less and 92% of runs had a 2% error or less for scale normalization. The nonlinear calibration functions used in the simulation are shown in Figure 6.5.

Note that when the identity calibration function is imposed, there is no need for normalization. From this limited simulation experiment it appears that invariant set normalization is superior to both quantile and qspline normalization methods. Invariant set normalization has smaller errors (for unchanged genes) for all scenarios considered here. Note that for a small number of probes, the qspline normalized values were negative. The percentage of negative values (per run of the simulation) was always less than 1%. Negative values were generated under the nonlinear calibration functions and identity calibration with $\lambda = 4$, but not with $\lambda = 2$. Based on this simulation study and the rationale behind the methods, we recommend the use of invariant set normalization.

Figure 6.3: The empirical cdf functions for the scenarios with $\lambda = 2$ and $\lambda = 4$ with **identity calibration function** and no error.

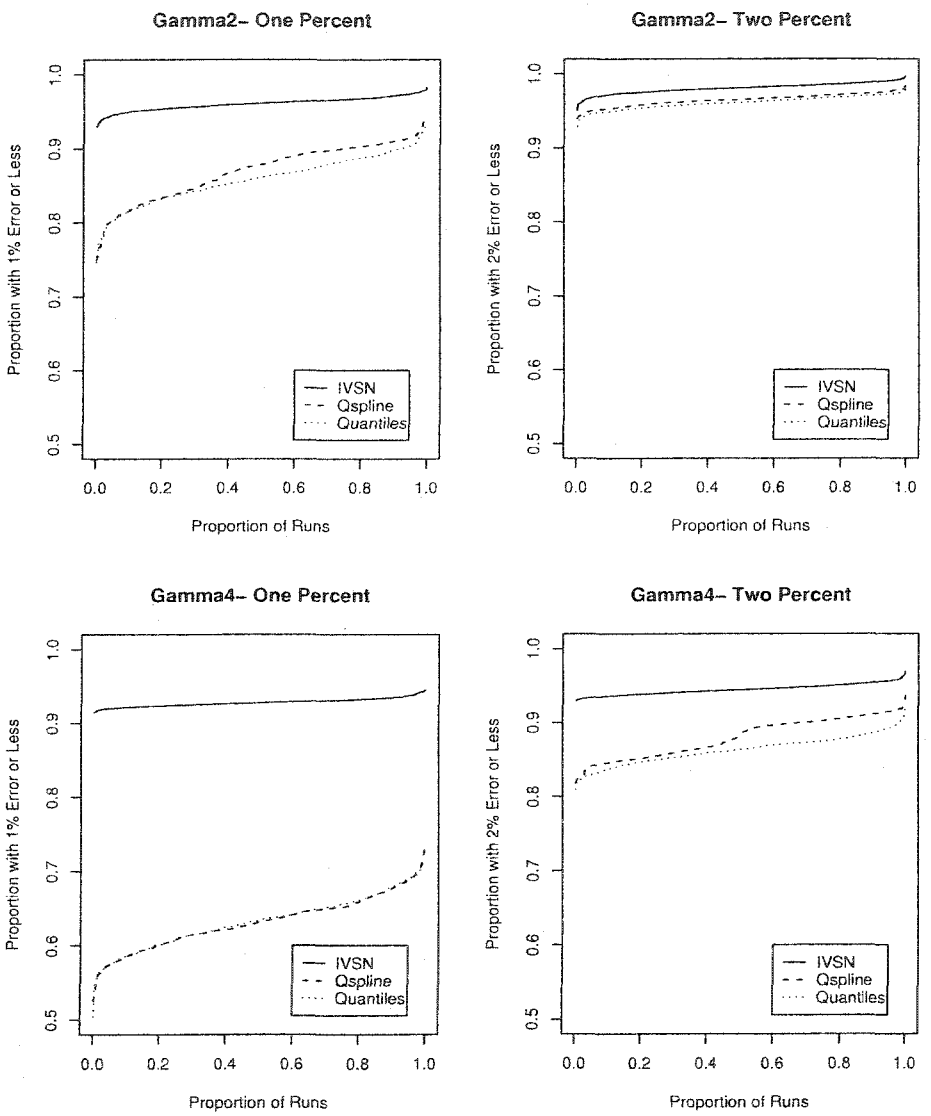


Figure 6.4: The empirical cdf functions for the scenario with $\lambda = 2$ with **nonlinear calibration functions** and no error.

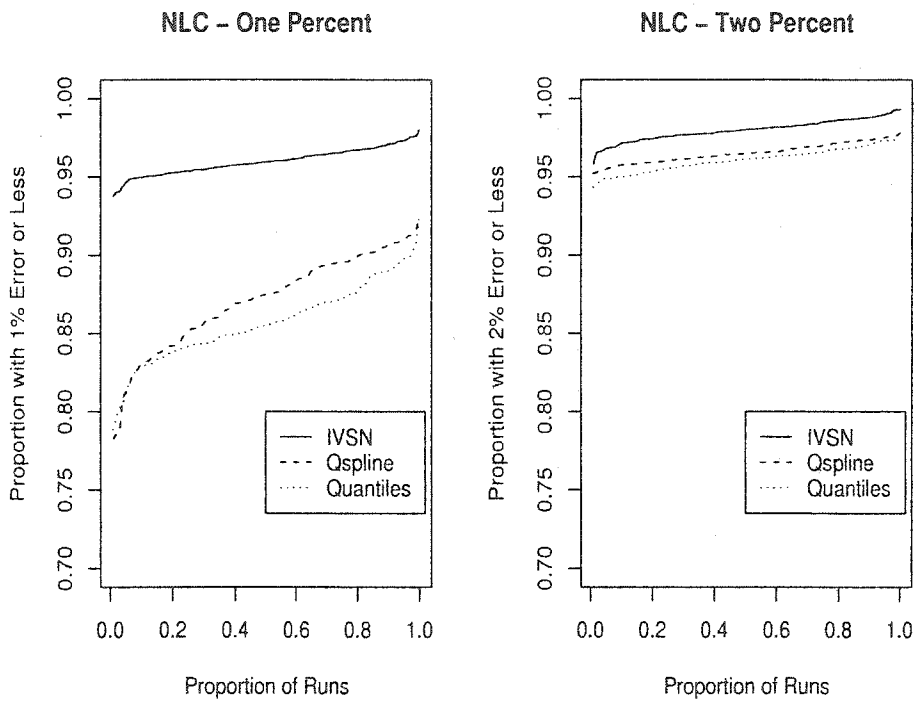
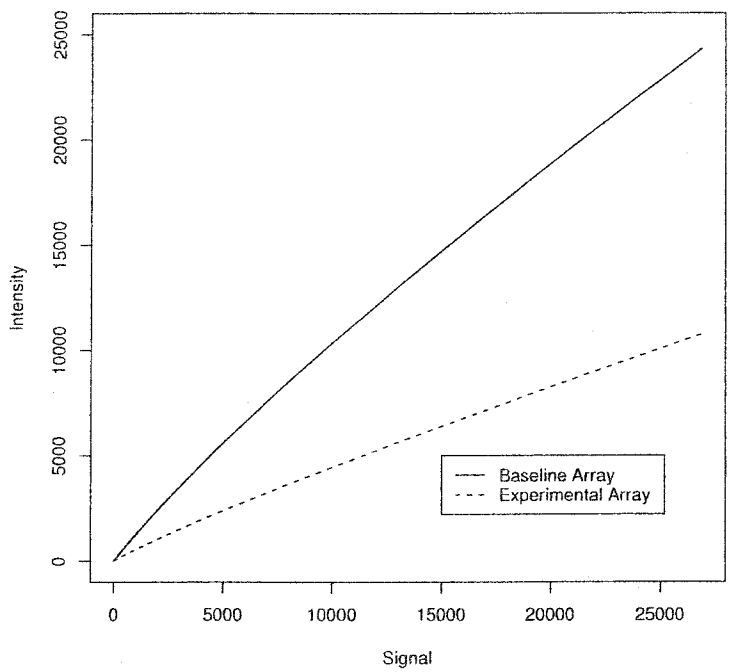


Figure 6.5: The nonlinear calibration functions used in the simulation.



Chapter 7

A UNIFIED MODEL FOR OLIGO ARRAYS

Currently, most models for oligo arrays start with preprocessed data. Namely, background correction, normalization and nonspecific binding corrections are made separately and before calculation of an expression index. Each correction is assumed to be known and a measure of uncertainty is only calculated at the expression index step. This process makes it difficult to calculate an overall measure of uncertainty for the final computed index. A unified model would is proposed here which incorporates all of the preprocessing steps with the expectation that it will allow for a full likelihood based analysis rather than a sequence of heuristic analyses. Our proposed model includes most of the current commonly used models as special cases.

7.1 Literature Review

Very recently a unified model has been proposed by Wu *et al.* [76]. They propose the following model:

$$\begin{aligned} PM_{ijkn} &= O_{ij} + N_{ijkn} + S_{ijkn} \qquad (7.1) \\ &= O_{ij} + exp(\mu_{ijkn} + \varepsilon_{ijkn}) + exp(s_n + \delta_n X_{ij} + a_{ijkn} + b_{ij} + \xi_{ijkn}), \end{aligned}$$

for the kth probe pair of the nth probe set for the jth replicate of the ith treatment. They assume O_{ijkn} represents optical noise and follows a log-normal distribution, μ_{ijkn} represents nonspecific binding, ε_{ijkn} is normally distributed and accounts for the amount of nonspecific binding for the same probe varying from array to array, s_n represents baseline log expression for probe set n, δ_n is the expected differential

expression for a unit difference in the variable X_i (where X_i would commonly be an indicator variable for treatment i), a_{ijkn} represents probe affinity, b_{ij} is a scale calibration term, ξ_{ijkn} is a normally distributed error term. Note that δ_n is the parameter of interest. Also note that a_{ijkn} and μ_{ijkn} are functions of α (a calculated probe affinity term). This unified model only allows for a scale calibration incorporated into the specific binding term (S_{ijkn}). Previously, Irizarry and coauthors recommended the use of quantile normalization when compared to scale normalization and other methods [11].

7.2 Proposed Unified Model

Before we consider a unified model, let us consider what is common between the major models: MAS (Equation 3.1), RMA (Equation 3.4), GC-RMA (Equations 3.5 and 3.6), MBEI (Equations 3.2 and 3.3) and the Chu *et al.* (Equations 3.7-3.9). To do this, we consider each of the models on the original (not log) scale. All of the models considered here contain a probe affinity term (Φ) and an expression term (θ). These are the only terms common to all models.

Most of the models contain a multiplicative error term, but the MBEI model contains only an additive error term. The presence of both multiplicative and additive error terms is suggested by the Rocke model (Equation 3.10). A term that is unique to the Chu *et al.* models is the δ term. This term seems to account for the fact that the available transcript abundances for each array may not be the same. In other words, replicates of the same treatment may not have the same amount of transcript available for binding. This is different from the multiplicative error term that suggests that even if the same amount of transcript was available for two replicates, the amount that binds would vary.

Nonspecific binding is incorporated into most models, but in very different ways. The PM-MM version of MBEI and some of the Chu *et al.* models adjust for optical

background and NSB using MM. MAS uses the IM term to account for nonspecific binding. RMA and MBEI take an empirical approach to account for nonspecific binding. NSB is estimated after normalization using the MBEI PM-only model. Optical noise and NSB are combined into a single term for the RMA model, which is estimated prior to normalization. GC-RMA (and the Wu unified model) use a calculated probe affinity (α) to account for nonspecific binding.

A calibration function relates observed intensity to signal (including both GSB and NSB), while normalization attempts to correct for the calibration function in order to allow for array to array comparisons. This implies that NSB should be estimated after normalization.

Considering the features of each of these models, we propose the following unified model:

$$PM_{ijkn} = b_{ij} + f_{ij}(\nu_{ijkn} + \Phi_{kn}\theta_{in}2^{\delta_{ijn}+\eta_{ijkn}}) + \varepsilon_{ijkn} \tag{7.2}$$

$$MM_{ijkn} = b_{ij} + f_{ij}(\nu'_{ijkn} + \phi_{kn}\theta_{in}2^{\delta_{ijn}+\eta'_{ijkn}}) + \varepsilon'_{ijkn}, \tag{7.3}$$

where δ_{ijn}, η_{ijn} and ε_{ijkn} are independently distributed error terms. The calibration function (f) expressed the relationship between the amount of binding that has taken place versus the intensity of a probe. The overall optical background for an array is modelled as b, but will vary from probe to probe by some additive error (ε). The nonspecific binding is represented by ν.

Let us compare our proposed unified model (Equations 7.2 and 7.3) to the unified model proposed by Wu *et al.* (Equation 7.1). One primary difference is the form of the normalization or calibration function. We have a single calibration function acting on the sum of the gene specific and nonspecific signal. In comparison, Wu *et al.* allow for two separate normalization functions, one for nonspecific signal and another for specific signal.

7.3 Estimability of the Unified Model (in the absence of random errors)

Starting with the simplest case, let us consider the observed PM and MM values in the absence of random error and with a scale calibration function:

$$PM_{ijkn} = b_{ij} + f_{ij} \times (\nu_{ijkn} + \Phi_{kn}\theta_{in}) \tag{7.4}$$

$$MM_{ijkn} = b_{ij} + f_{ij} \times (\nu'_{ijkn} + \phi_{kn}\theta_{in}). \tag{7.5}$$

The primary goal is to estimate the fold changes $\theta_{in}/\theta_{i'n}$, where i and i' are two different treatments. There are two possible assumptions that will allow us to estimate fold change from this unified model.

Assumption 1: The amount of nonspecific binding will be the same for PM and MM members of the same probe pair. If this is the case, then $\nu_{ijkn} = \nu'_{ijkn}$. Then we consider the PM-MM ratios:

$$\frac{PM_{e\cdot kn} - MM_{e\cdot kn}}{PM_{b\cdot kn} - MM_{b\cdot kn}} = \frac{f_e \times (\Phi_{kn} - \phi_{kn})\theta_{en}}{f_b \times (\Phi_{kn} - \phi_{kn})\theta_{bn}} \tag{7.6}$$

$$= \frac{f_e \times \theta_{en}}{f_b \times \theta_{bn}}, \tag{7.7}$$

where $PM_{b\cdot kn}$ and $MM_{b\cdot kn}$ represent the kth probe pair of the nth probe set for a baseline array and $PM_{e\cdot kn}$ and $MM_{e\cdot kn}$ represent the kth probe pair of the nth probe set for an experimental array.

If we assume that most genes are not differentially expressed, then we can estimate the relative calibration scales $(\frac{f_e}{f_b})$. For a gene that is not differentially expressed, $\theta_{en} = \theta_{bn}$. Hence for these genes,

$$\frac{PM_{e\cdot kn} - MM_{e\cdot kn}}{PM_{b\cdot kn} - MM_{b\cdot kn}} = \frac{f_e}{f_b}.$$

So we can estimate $\frac{f_e}{f_b}$ as the mode of the empirical distribution of ALL PM-MM ratios. Hence we can estimate the FC as

$$\frac{f_b(PM_{e\cdot kn} - MM_{e\cdot kn})}{f_e(PM_{b\cdot kn} - MM_{b\cdot kn})} = \frac{\theta_{en}}{\theta_{bn}} = \hat{FC}.$$

Assumption 2: MM probes are not picking up any gene specific signal. If we assume that the (gene specific) affinity is zero for MM probes then $\phi_{kn} = 0$. This assumption allows us to use the affinity terms (α) computed based on probe sequence. If we assume that the function h relates affinity to NSB, then $\nu_{ijkn} = h_{ij}(\alpha_{kn})$ and $\nu'_{ijkn} = h_{ij}(\alpha'_{kn})$, where α is assumed known. If we additionally assume that optical background (b_{ij}) can be estimated separately, then the background corrected MM values can be expressed as:

$$MM_{ijkn} - \hat{b}_{ij} = f_{ij} \times h_{ij}(\alpha'_{kn}).$$

Hence we can estimate the composite function $f_{ij}h_{ij}$ (relating affinity to intensity due to NSB) for each array ij from MM values. Then we can "correct" the PM values (for optical noise and NSB) by taking

$$PM^c_{ijkn} = PM_{ijkn} - \hat{b}_{ij} - \hat{f}_{ij}\hat{h}_{ij}(\alpha_{kn}) = f_{ij}\hat{\Phi}_{kn}\theta_{in}.$$

Note that we can only perform the NSB correction in this way if we assume a scale calibration function.

If we again assume that most genes are not differentially expressed, then we can estimate the relative calibration scales. For a gene that is not differentially expressed, $\theta_{en} = \theta_{bn}$. Hence for these genes,

$$\frac{PM^c_{e\cdot kn}}{PM^c_{b\cdot kn}} = \frac{f_e}{f_b}.$$

So we can estimate $\frac{f_e}{f_b}$ as the mode of the empirical distribution of ALL PM^c ratios. Hence we can estimate the FC as

$$\frac{f_b PM^c_{e\cdot kn}}{f_e PM_{b\cdot kn}} = \frac{\theta_{en}}{\theta_{bn}} = \hat{FC}.$$

Note: Neither of these assumptions is likely to be true and it is unclear which leads to more accurate estimates of fold change.

Chapter 8

PROBE LEVEL DIAGNOSTICS AND A TEST FOR DIFFERENTIAL EXPRESSION

This chapter presents some probe level diagnostics for microarray data. In addition we present a method for combining probe level tests of differentially expression. We will focus primarily on preprocessed (background corrected, normalized and NSB corrected) data. We will consider RMA and dChip models separately.

RMA: Based on the RMA model (Equation 3.4) in the absence of random errors a preprocessed PM value (denoted as PM^*) on the log_2 scale is:

$$log_2(PM^*_{ijkn}) = log_2 N(B(PM_{ijkn})) = log_2(\Phi_{kn}) + log_2(\theta_{ijn}). \tag{8.1}$$

dChip: Based on the dChip model (Equations 3.2 and 3.3) in the absence of random errors a preprocessed PM value (denoted as PM') is:

$$PM'_{ijkn} = B(N(PM_{ijkn})) = \Phi_{kn}\theta_{ijn} \tag{8.2}$$

and a preprocessed PM-MM difference (denoted as PM") is:

$$PM''_{ijkn} = N(PM_{ijkn}) - N(MM_{ijkn}) = (\Phi_{kn} - \phi_{kn})\theta_{ijn}. \tag{8.3}$$

In other words, for either model, after preprocessing we should be left with signal. We will consider a number of diagnostics, some of which are customized to the RMA and dChip models. Similar diagnostics could be developed for other models and methods.

The following questions are considered.

1. Was the preprocessing of the data successful? If the original model is correct and preprocessing is successful, then the resulting data should follow a simple model involving only signal (abundance $\times$ affinity). Combined with the assumption that a large number of genes are unchanged we argue that the mode of the distribution of the elementary probe level fold change estimates should equal one on the raw scale or zero on the log scale. This can be checked using observed data.

2. How consistent is the probe level information based on concordance among probe level fold change estimates within a probe set? For a given probe set we consider the difference between the estimated log FC by probe and the overall estimated log FC for the probe set.

3. How consistent is the probe level information based on concordance of probe level p-values within a probe set?

8.1 Examination of Background Corrected and Normalized Data

It is typically expected that the majority of genes will be "unchanged". This implies that the majority of the estimated fold changes will be one (on the original scale) or zero (on the log scale). In this section, we consider the elementary (probe level) estimated fold changes. These values can be easily obtained after preprocessing. We can then examine their distribution, to see if the mode is in fact one (on the original scale) or zero (on the log scale). If so, this is an indication that the normalization and background correction have been successfully performed.

RMA: For RMA model, in the absence of random errors, the difference of the logged preprocessed values (Equation 8.1) provides an estimate of log fold change

$$log_2(PM^*_{ijkn}) - log_2(PM^*_{i'j'kn}) = log_2\left(\frac{\theta_{ijn}}{\theta_{i'j'n}}\right) = log_2(\hat{FC}). \qquad (8.4)$$

Assuming that the errors are generally small compared to the signal, then the differences (of the logged values) should be close to zero for unchanged genes. If we also assume that the majority of genes are not differentially expressed, then in the absence of error the mode of differences would be exactly zero.

dChip: For the MBEI model, in the absence of random errors, the ratio of the preprocessed values (Equation 8.2) provides an estimate of the fold change

$$\frac{PM'_{ijkn}}{PM'_{i'j'kn}} = \frac{\theta_{ijn}}{\theta_{i'j'n}} = \hat{FC} \tag{8.5}$$

and the ratio of the differenced preprocessed values (Equation 8.3) also provide and estimate of the fold change

$$\frac{PM''_{ijkn}}{PM''_{i'j'kn}} = \frac{\theta_{ijn}}{\theta_{i'j'n}} = \hat{FC}. \tag{8.6}$$

Assuming that the errors are generally small compared to the signal, then the ratios should be close to one for unchanged genes. If we also assume that the majority of genes are not differentially expressed, then in the absence of error the mode of the ratios would be exactly one.

Effect of error on the mode of the ratios: We examine the effect of error on the mode of the ratios. Although Li and Wong represent the additive error as independent from the signal [41], we have noted that the error is proportional to the signal. (Refer to Figure 5.4 and discussion of that figure.) So, after preprocessing we consider,

$$\frac{PM'_{ijkn}}{PM'_{i'j'kn}} = \frac{\theta_{ijn}(1+\rho\varepsilon_{ijkn})}{\theta_{i'j'n}(1+\rho\varepsilon_{i'j'kn})}. \tag{8.7}$$

Our primary interest is in unchanged genes (since the mode should be unaffected changed genes), for which $\theta_{ijn} = \theta_{i'j'n}$ and this quantity reduces to

$$\frac{1+\rho\varepsilon_1}{1+\rho\varepsilon_2},$$

where we assume $\varepsilon_i \sim N(0,1)$ for $i = 1, 2$. We examine the distribution of this ratio by simulation for $\rho = 0.01, 0.1$ and 0.5 in Figure 8.1. These graphs were created by generating 100,000 realization from a standard normal for each ε_1 and ε_2. From this we see that the error has no effect on the mode of the distribution, although the symmetry is affected.

In order to implement this diagnostic in practice, we need a method to estimate the mode. For the difference of the logged preprocessed PM values (Equation 8.4), we estimate the mode (m) as the value that maximizes:

$$\sum_{kn} I(|log(PM^*_{ijkn}) - log(PM^*_{i'jkn}) - \hat{m}| < \varepsilon) \tag{8.8}$$

For the ratios of the preprocessed PM values (Equations 8.2 and 8.3), we estimate the mode (m) as the value that maximizes:

$$\sum_{kn} I(|\frac{PM'_{ijkn}}{PM'_{i'jkn}} - \hat{m}| < \varepsilon) \tag{8.9}$$

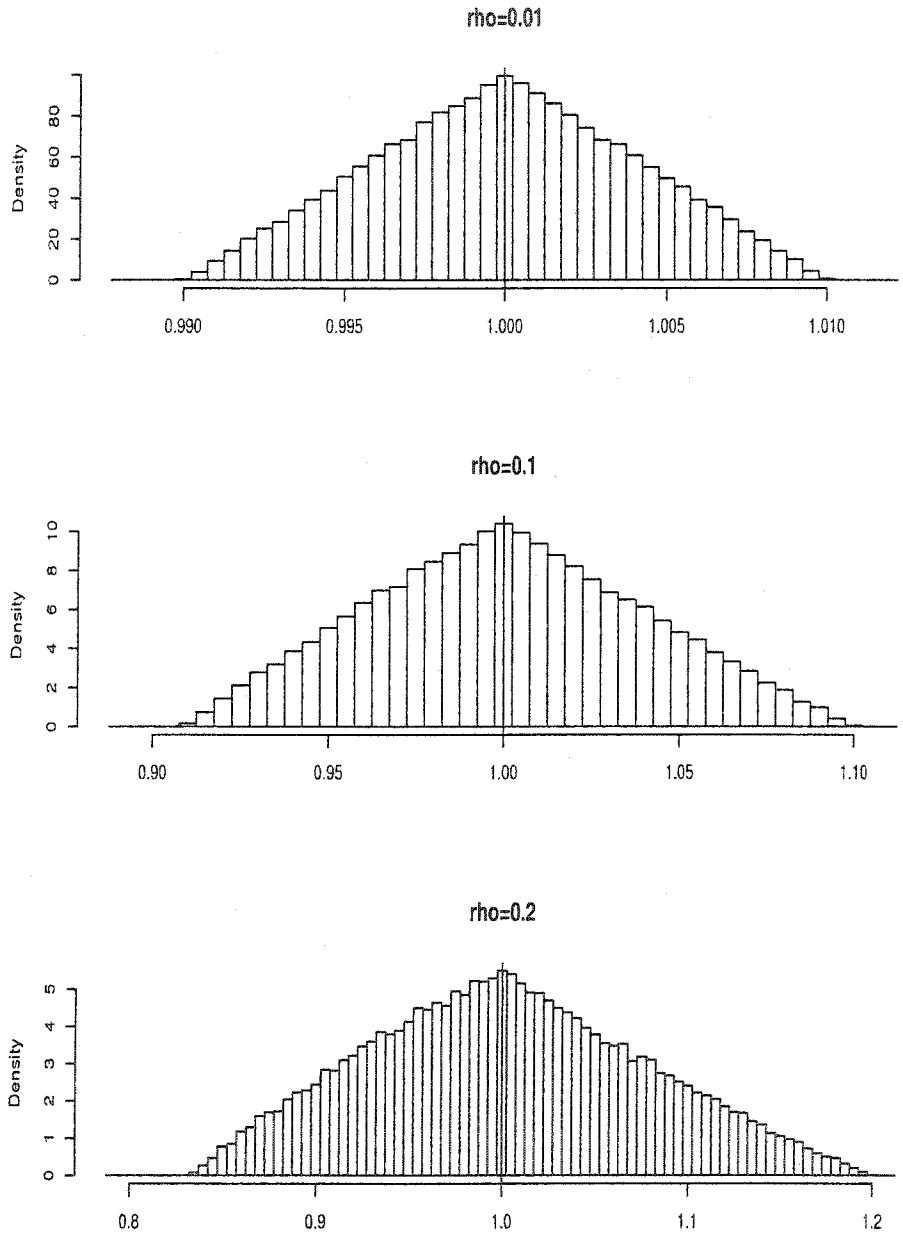
or

$$\sum_{kn} I(|\frac{PM''_{ijkn} - MM''_{ijkn}}{PM''_{i'jkn} - MM''_{i'j'kn}} - \hat{m}| < \varepsilon). \tag{8.10}$$

The estimated mode will depend on the selected value of ε. This is a quick method for estimating the mode. Vieu [71] presents a discussion of methods for density mode estimation.

Example 1: Let us consider four barley1 arrays representing Legacy (Arrays 11 and 12) and Merit (Arrays 21 and 22) cultivars. We preprocess the data using RMA and dChip. For dChip, all arrays are normalized to the second Merit array (Array22). In addition to the overall estimated mode, we consider the estimated mode by quartile. If we assume that most genes (in any given quartile) are not differentially expressed, then the mode of the ratios by quartile should also equal one. The quartiles are defined by the preprocessed intensities for Array22. All differences and ratios are relative to Array22.

Figure 8.1: Simulated distribution of the ratios (for unchanged genes) with error for ρ=0.01, 0.1 and 0.5.



For the purposes of comparison, we consider the differences of the logged values and the ratios on the original scale of the preprocessed PM values from **both RMA and dChip**. We examine the estimated modes of the differences of the logged preprocessed PM values for RMA (PM^*) and dChip (PM'). Values less than or equal to zero are set to 0.01. The modes are shown in Table 8.1. We also examine the estimated modes of the ratios of the preprocessed PM values for RMA (PM^*) and dChip (PM'). To prevent undefined values, we set preprocessed values of zero to 0.01 on Array22 only. The modes are shown in Table 8.2. Finally we consider the estimated modes of the ratios of the preprocessed $PM'' - MM''$ differences from dChip. These modes are shown in Table 8.3.

Table 8.1: Estimated modes ($\varepsilon = 0.02$) of the differences of the preprocessed $log_2(PM)$ values (Equation 8.4).

	Array11			Array12			Array21		
	Raw	RMA	dChip	Raw	RMA	dChip	Raw	RMA	dChip
overall	-0.10	0.10	0.01	0.00	0.04	-0.01	0.07	0.01	0.01
Q1	-0.16	0.01	0.01	0.00	-0.01	-0.01	-0.07	0.04	0.01
Q2	-0.08	0.15	0.01	0.05	0.13	0.07	0.04	0.00	0.00
Q3	-0.01	0.10	0.00	-0.01	0.01	-0.01	0.10	0.01	0.01
Q4	0.01	0.04	-0.01	-0.02	0.04	0.01	0.17	0.00	0.00

Table 8.2: Estimated modes ($\varepsilon = 0.02$) of the ratios of the preprocessed PM values (Equation 8.5).

	Array11			Array12			Array21		
	Raw	RMA	dChip	Raw	RMA	dChip	Raw	RMA	dChip
overall	0.92	1.00	0.99	0.99	1.00	0.99	1.03	1.00	0.99
Q1	0.87	1.00	0.99	0.99	0.97	0.99	0.95	1.00	1.01
Q2	0.91	0.99	0.95	0.99	1.00	0.99	1.01	0.99	0.99
Q3	0.94	0.99	1.00	0.97	1.00	0.99	1.06	1.00	1.00
Q4	0.99	1.03	1.00	0.94	0.97	0.99	1.10	0.99	1.00

Figure 8.2: Histograms of log differences for Arrays 11 and 12 for the second quartile. The estimated mode is marked by a vertical line.

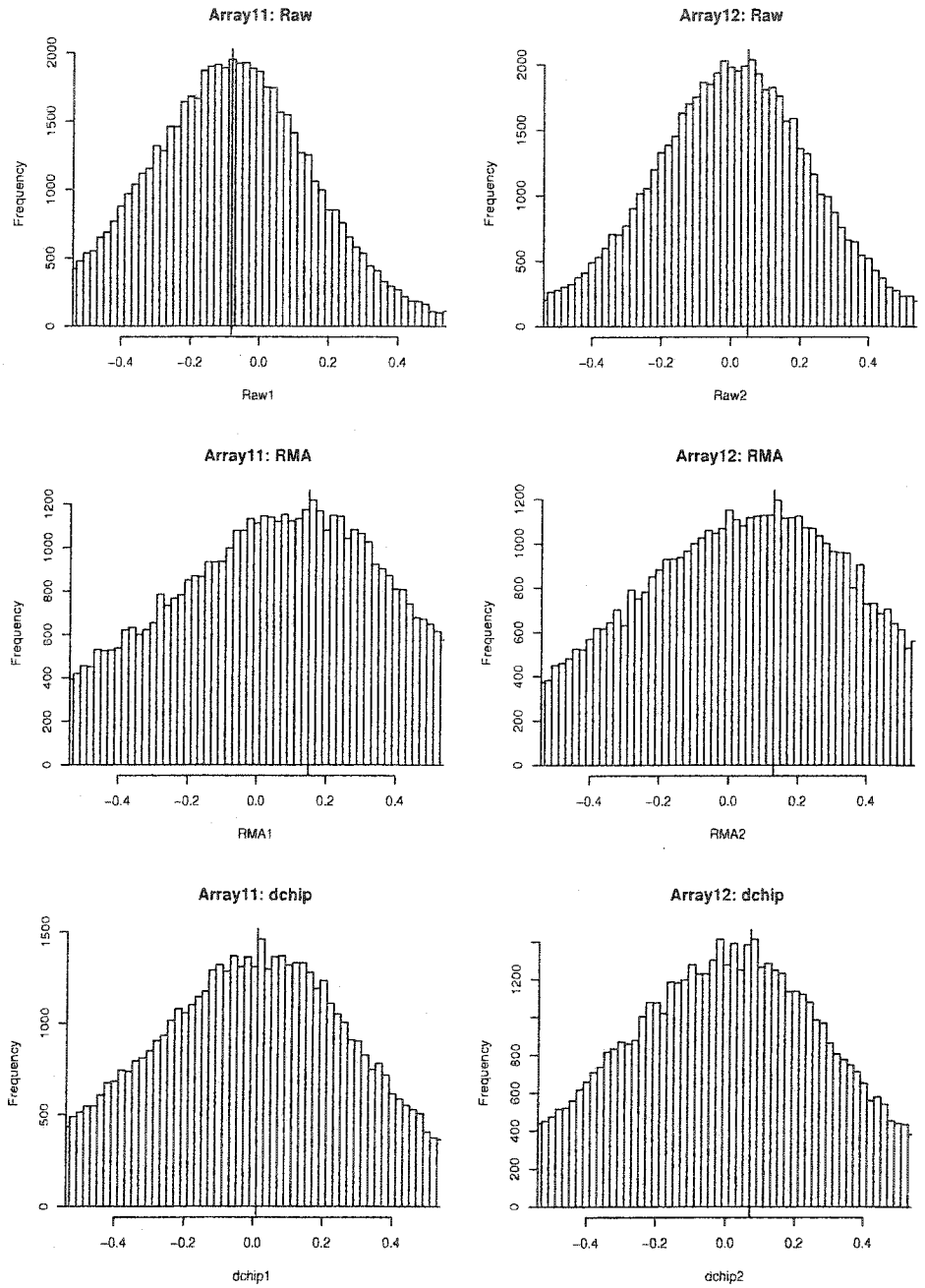


Table 8.3: Estimated modes ($\varepsilon = 0.02$) of the ratios of the preprocessed PM-MM values (Equation 8.6).

	Array11		Array12		Array21	
	Raw	dChip	Raw	dChip	Raw	dChip
overall	0.91	0.99	0.89	0.89	1.09	0.99
Q1	1.00	0.99	1.01	1.01	0.99	0.99
Q2	0.91	0.94	0.89	0.95	0.95	0.93
Q3	0.91	0.91	0.92	0.87	1.09	0.97
Q4	0.97	0.95	0.87	0.94	1.11	0.95

Note that for the $log_2(PM)$ differences (Table 8.1) the RMA and dChip background corrected and normalized data do not always offer an improvement over the raw data. For example, the mode of the RMA adjusted values is further from zero than the raw values for Arrays 11 and 12 for the second quartile. This is also true for dChip adjusted values for Array 11 for the second quartile. Histograms of log differences for Arrays 11 and 12 for the second quartile are shown in Figure 8.2.

It is interesting to note that the "shifted modes" we observed on the log scale are much less drastic when we look at the ratios on the original scale. For the PM ratios (Table 8.2), RMA adjusted values usually (but not always) perform as well as or better than the raw data. The dChip adjusted PM values always perform as well as or better than the raw data. However, the performance of the dChip adjusted PM-MM values do not always offer an improvement in the estimated mode. This could be due to the additional assumptions (such as identical background and nonspecific binding for both members of a probe pair) required when using PM-MM.

8.2 Consistency of the Probe Level Fold Change Estimates

We once again consider the probe level estimates of fold change, which can be easily computed after preprocessing. For a given probe set, each PM probe or probe pair can be used to estimate the same quantity, namely the relative transcript abundance for the gene corresponding to the probe set. Since this is the case, we

can check the consistency of these probe level estimates from a given probe set. For this purpose we consider the deviation of the probe level logFC from the overall estimated logFC for the probe set:

$$log_2(\hat{FC}_{kn}) - log_2(\hat{FC}_n), \tag{8.11}$$

where $log_2(\hat{FC}_{kn}) = log_2(\frac{PM^*_{ijkn}}{PM^*_{i'j'kn}})$ for RMA, $log_2(\hat{FC}_{kn}) = log_2(\frac{PM'_{ijkn}}{PM'_{i'j'kn}})$ or $log_2(\frac{PM''_{ijkn}}{PM''_{i'j'kn}})$ for dChip and $log_2(\hat{FC}_n)$ is the median (for the probe set) of the $log_2(\hat{FC}_{kn})$ values. We would like to choose some bounds for these deviations, such that if a probe exceeds these bounds, we could flag it as a potential outlier. In order to choose these bounds, we consider all of the possible deviations for a pair of arrays (or all possible pairs of arrays). From this distribution we can choose the 5th and 95th percentiles as our bounds, and any probe (or probe pair) that exceeds these bounds will be flagged as an outlier. A flagged probe is giving an estimated fold change that is discordant with other probes in the same probe set.

In practice this diagnostic can be used to identify outlier probes. In addition, if an investigator was concerned about a particular gene, this diagnostic could be used to consider why the gene was identified or failed to be identified as differentially expressed.

Example 2: We once again use the barley data, but consider only a single replicate for each Legacy and Merit. The barley1 array has 22840 probe sets typically each represented by 11 probe pairs. Here we apply the proposed method to **RMA corrected data only**. After background correction and normalization as prescribed by the RMA algorithm, we computed all of the deviations for all probes and all probe sets. The histogram of these values are shown in Figure 8.3a. The 5th percentile was found to be -0.77 and the 95th percentile was 0.61. A probe is flagged if its deviation exceeds these bounds.

Considering only those probe sets with exactly 11 probe pairs, a histogram of the number of flagged probes per probe set is shown in Figure 8.3b. Note that the vast majority of probe sets have three or fewer probes flagged as outliers.

Here we consider just two examples. First we consider probe set 424 which has no flagged probes. A scatter plot of the Legacy vs. Merit $log_2(PM)$ values for probe set 424 (Contig10213-x-at) are shown in Figure 8.4. The estimated log fold changes and deviations by probe are shown in Table 8.4. Note that none of the deviations exceed the bounds and there are no obvious outliers in the scatter plot. For this probe set $log_2(\hat{FC}_{424}) = 0.018$.

We also consider probe set 328 (Contig10101-at) which has a single flagged probe. A scatter plot of the $log_2(PM)$ values for probe set 328 are shown in Figure 8.5. The estimated log fold changes and deviations by probe are shown in Table 8.5. Probe 8 is an obvious outlier on the scatter plot in Figure 8.5 and has been flagged based on its deviation from the overall probe set estimated log FC. For this probe set $log_2(\hat{FC}_{328}) = 0.148$, while for probe 8 $log_2(\hat{FC}_{8,328}) = -3.600$.

Note that overlaid on each of the scatter plots is the line:

$$log_2(PM_{Legacy}) = log_2(\hat{FC}_n) + log_2(PM_{Merit}),$$

where $log_2(\hat{FC}_n)$ is the median of the log fold changes for the probe set.

Figure 8.3: (a) Histogram of the deviations for all probes and all probe sets (with 5th and 95th percentiles marked) for Example 2. (b) Histogram of the number of flagged probes per probe set. Here we consider only the 22801 probe sets with 11 probe pairs.

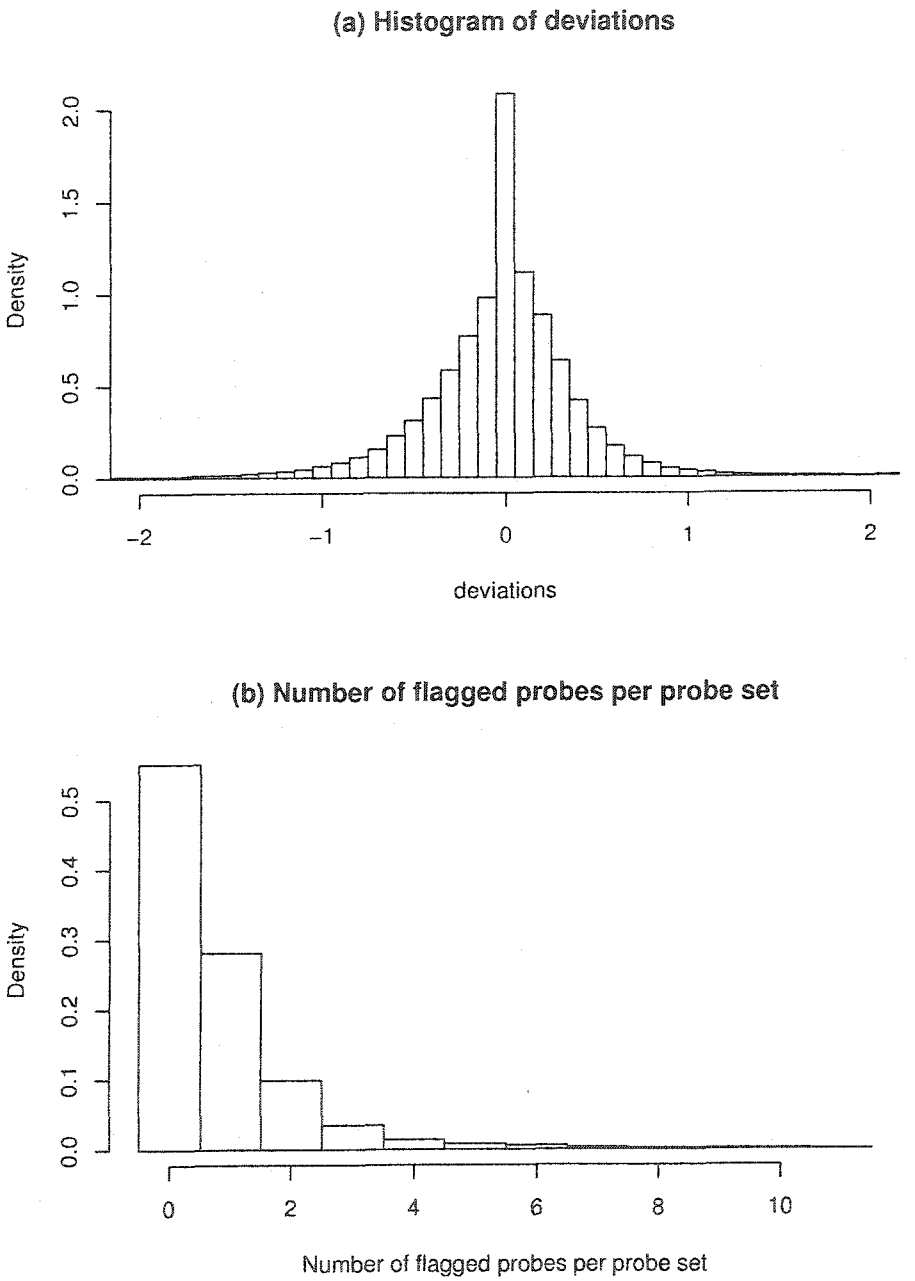


Figure 8.4: Scatter plot of the $log_2(PM)$ values for probe set 424.

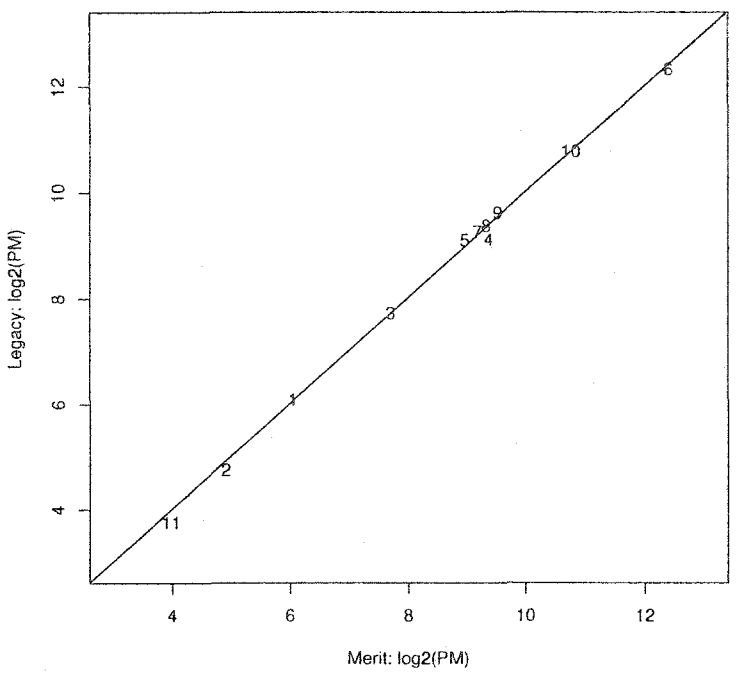


Table 8.4: Estimated logFCs and deviations by probe for probe set 424 ($log_2(\hat{FC}_{424}) = 0.018$).

probe	$log_2(\hat{FC}_{kn})$	deviation	flag
1	0.042	0.024	0
2	-0.160	-0.170	0
3	0.018	0.000	0
4	-0.260	-0.280	0
5	0.140	0.120	0
6	-0.067	-0.085	0
7	0.082	0.064	0
8	0.034	0.016	0
9	0.090	0.072	0
10	0.018	-0.000	0
11	-0.230	-0.240	0

Figure 8.5: Scatter plot of the $log_2(PM)$ values for probe set 328.

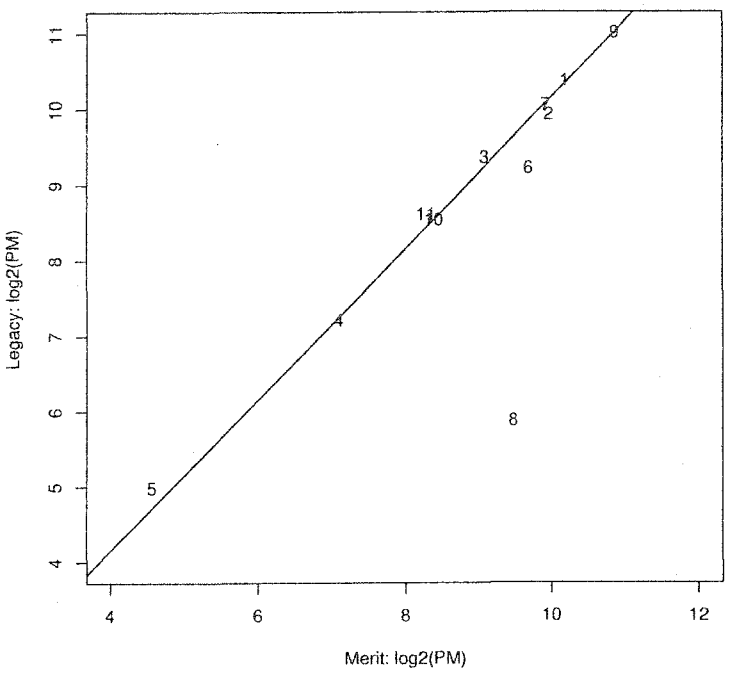


Table 8.5: Estimated logFCs and deviations by probe for probe set 328 ($log_2(\hat{FC}_{328}) = 0.148$).

probe	$log_2(\hat{FC}_{kn})$	deviation	flag
1	0.190	0.042	0
2	-0.042	-0.190	0
3	0.280	0.140	0
4	0.120	-0.027	0
5	0.410	0.270	0
6	-0.470	-0.610	0
7	0.120	-0.025	0
8	-3.600	-3.700	1
9	0.150	0.000	0
10	0.160	0.009	0
11	0.330	0.180	0

8.3 A Test for Differential Expression Using Combined Probe Level p-values

Currently detection of differentially expressed genes is based on expression index values which provide an expression value by gene and array. In other words, the probe level information is combined into a summary value by gene (and array). It is possible to test for differential expression using information from a single (PM) probe. Assuming we have multiple arrays for two treatment groups, we could use any two sample test on the preprocessed values for a single probe across treatments. Each PM probe in a given probe set can be used to estimate the relative transcript abundance for the gene corresponding to that probe set. This suggests that we could combine information from the p-values from multiple probes within a probe set.

Fisher proposed a method for combining p-values from independent tests of significance [22]. Here we consider an approach for combining dependent p-values. We would like to combine probe level p-values. However, all probes of the same probe set can be used to estimate the same transcript abundance and appear on the same arrays. So there is dependence among the p-values of the same probe set.

Assuming the RMA model (Equation 3.4) (or assuming an error proportional to signal for the dChip model) for a fixed probe set implies:

$$log_2(PM^*_{ijk}) = Y_{ijk} = \mu + P_k + T_i + e_{ijk},$$

where

$$e_{ijk} = \eta_{ij} + \varepsilon_{ijk}$$

and PM^* is a preprocessed PM value, P_k is a fixed probe effect, T_i is a treatment effect, η_{ij} is a random array effect with mean zero and variance $\sigma^2_{\eta_i}$, and ε_{ijk} is an error term with mean zero and variance $\sigma^2_{\varepsilon_i}$. We assume that the η_{ij}'s and ε_{ijk}'s are independent.

We would like to test for a difference between the average probe values for the baseline and experimental treatments by probe. The probe values from the baseline and treatment arrays are independent samples. We will work with the t-test, but any two sample test can be used. Let $\bar{Y}_{e.k}$ be the average (of the logged values) for probe k for those arrays representing the experimental treatment and let $\bar{Y}_{b.k}$ be the average (of the logged values) for probe k over those arrays representing the baseline (or control) treatment. Let $s_{e.k}$ be the standard deviation (of the logged values) for probe k over those arrays arrays representing the experimental treatment and $s_{b.k}$ is the standard deviation (of the logged values) over those arrays representing the baseline (or control) treatment. Finally, let n_e represent the number of experimental arrays and n_b represent the number of baseline arrays.

Equal Variances: Let us begin by considering the t-test assuming a common variance for both the baseline and experimental populations. We assume that $\eta_{ij} \sim N(0, \sigma_\eta^2)$ and $\varepsilon_{ijk} \sim N(0, \sigma_\varepsilon^2)$. Then the test statistic of interest is:

$$X_k = \frac{\bar{Y}_{e.k} - \bar{Y}_{b.k}}{\sqrt{s_k^2(\frac{1}{n_b} + \frac{1}{n_e})}}.$$

where $s_k^2 = \left(\frac{n_b - 1}{n_b + n_e - 2}\right) s_{b.k}^2 + \left(\frac{n_e - 1}{n_b + n_e - 2}\right) s_{e.k}^2$.

Note that

$$Var(Y_{bjk}) = Var(Y_{ejk}) = Var(e_{ijk}) = \sigma_e^2 = \sigma_\eta^2 + \sigma_\varepsilon^2.$$

Also,

$$\bar{Y}_{e.k} = \mu + P_k + T_e + \bar{e}_{e.k}$$

and

$$\bar{Y}_{b.k} = \mu + P_k + T_b + \bar{e}_{b.k}.$$

So,

$$\bar{Y}_{e.k} - \bar{Y}_{b.k} = T_e - T_b + \bar{e}_{e.k} - \bar{e}_{b.k}.$$

Let p_k be the p-value for the test for probe k. Then

$$\begin{aligned} p_k &= 2T_{n_b+n_e-2}\left(\frac{-|\bar{Y}_{e.k}-\bar{Y}_{b.k}|}{\sqrt{s_k^2(\frac{1}{n_b}+\frac{1}{n_e})}}\right) \\ &= 2T_{n_b+n_e-2}\left(\frac{-|\delta+\bar{e}_{e.k}-\bar{e}_{b.k}|}{\sqrt{s_k^2(\frac{1}{n_b}+\frac{1}{n_e})}}\right) \end{aligned} \tag{8.12}$$

where $\delta = T_e - T_b$ and $T_{n_b+n_e-2}$ represents the t cumulative distribution function with $n_b + n_e - 2$ degrees of freedom.

The joint distribution of the e_{ijk} values for probe k is equal to the joint distribution of the $e_{ijk'}$ values for probe k'. Hence the distributions of p_k and p_k' will be the same, indicating that the $var(p_k) = var(p_{k'})$. Furthermore, the joint distribution of the e_{ijk} and $e_{ijk'}$ values is equal to the joint distribution of the $e_{ijk'}$ and $e_{ijk''}$ values, indicating the the correlation between p_k and $p_{k'}$ will be the same for any two probes k and k'.

Unequal Variances: We assume that $\eta_{ij} \sim N(0, \sigma^2_{\eta_i})$ and $\varepsilon_{ijk} \sim N(0, \sigma^2_{\varepsilon_i})$. The test statistic of interest is:

$$Z_k = \frac{\bar{Y}_{e.k} - \bar{Y}_{b.k}}{\sqrt{\frac{s_e^2}{n_e} + \frac{s_b^2}{n_b}}}.$$

Note that

$$Var(Y_{bjk}) = Var(e_{bjk}) = \sigma^2_{e_b} = \sigma^2_{\eta_b} + \sigma^2_{\varepsilon_b}$$

and

$$Var(Y_{ejk}) = Var(e_{ejk}) = \sigma^2_{e_e} = \sigma^2_{\eta_e} + \sigma^2_{\varepsilon_e}.$$

Let p_k be the p-value for the test for probe k. Then

$$\begin{aligned} p_k &\approx 2T_{\hat{\nu}}\left(\frac{-|\bar{Y}_{e.k}-\bar{Y}_{b.k}|}{\sqrt{\frac{s_e^2}{n_e}+\frac{s_b^2}{n_b}}}\right) \\ &= 2T_{\hat{\nu}}\left(\frac{-|\delta+\bar{e}_{e.k}-\bar{e}_{b.k}|}{\sqrt{\frac{s_e^2}{n_e}+\frac{s_b^2}{n_b}}}\right), \end{aligned} \tag{8.13}$$

where $\delta = T_e - T_b$ and $T_{\hat{\nu}}$ represents the t cumulative distribution function with Welch-Satterthwaite approximate degrees of freedom ($\hat{\nu}$) degrees of freedom calculated as

$$\hat{\nu} = \frac{\left(\frac{s_b^2}{n_b} + \frac{s_e^2}{n_e}\right)^2}{\frac{1}{n_b-1}\left(\frac{s_b^2}{n_b}\right)^2 + \frac{1}{n_e-1}\left(\frac{s_e^2}{n_e}\right)^2}.$$

Using the same logic as for the equal variance case, we can conclude that $var(p_k) = var(p_{k'})$ and the correlation between p_k and $p_{k'}$ is the same for any two probes k and k'.

Combining Dependent p-values: Let p_i be the p-value for the test using information about probe i. If $i = 1, ..., m$ (for a fixed probe set) and $s_i = -2ln(p_i)$, let us denote $corr(s_i, s_j) = c$ for $i \neq j$. Under the null hypothesis ($T_e = T_b$), s_i is distributed as a χ^2 variable with 2 degrees of freedom. Let

$$W = \sum_{i=1}^{m} \omega_i s_i$$

be the sum of the weighted statistics such that $\sum_{i=1}^{m} \omega_i = 1$. The null distribution of W may be approximated using a scaled χ^2 distribution. Specifically, suppose

$$\nu \frac{W}{E(W)} \sim \chi^2_\nu, \tag{8.14}$$

where the degrees of freedom ν are chosen as

$$\nu = 2\frac{E(W)^2}{var(W)},$$

using the approximation from [52]. Essentially this is a Satterthwaite moment based approximation.

We know

$$E(W) = E(\sum_{i=1}^{m} s_i \omega_i) = 2$$

and

$$Var(W) = \sum_{i=1}^{m} \omega_i^2 var(s_i) + \sum\sum_{i \neq j} \omega_i \omega_j cov(s_i, s_j) = 4\sum_{i=1}^{m} \omega_i^2 + \sum\sum_{i \neq j} 4c\omega_i\omega_j$$

since $cor(s_i, s_j) = c$ and $var(s_i) = 4$.

In order to implement this method, we need an estimate of c. Using the logic outlined by Makambi [44], we consider the quadratic form

$$q = \sum_{i=1}^{m} \frac{(s_i - \bar{s})^2}{(m-1)}.$$

This is the sample variance of the s_i values. We find that

$$\begin{aligned}
E(q) &= E\left(\frac{1}{m-1}\sum_{i=1}^{m}(s_i^2 - 2s_i\bar{s} + \bar{s}^2)\right) \\
&= \frac{1}{m-1}E\left[\sum_{i=1}^{m} s_i^2 - \frac{2}{m}\sum_{i=1}^{m}\sum_{j=1}^{m} s_i s_j + \frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{m} s_i s_j\right] \\
&= \frac{1}{m-1}\left[\frac{m-1}{m}\sum_{i=1}^{m} E(s_i^2) - \frac{1}{m}\sum_{i\neq j}\sum E(s_i s_j)\right] \\
&= \frac{1}{m-1}\left[\frac{m-1}{m}\sum_{i=1}^{m}(Var(s_i) + E(s_i)^2) - \frac{1}{m}\sum_{i\neq j}\sum(Cov(s_i, s_j) + E(s_i)E(s_j))\right] \\
&= \frac{1}{m-1}\left[\frac{8m(m-1)}{m} - \frac{4(m-1)(c+1)}{m}\right] \\
&= 4(1-c).
\end{aligned}$$

Then a method of moments estimator for c is

$$\hat{c} = 1 - q/4$$

where q is the sample variance of the s_i values. Since the correlations between any two s_i are the same and the variances of the s_i's are the same, we conjecture that equal weights will provide the optimum test. Hence we choose $\omega_i = 1/m$ and $W = \bar{s}$. Then

$$\hat{\nu} = 2\frac{E(W)^2}{var(W)} = \frac{8}{\frac{4}{m}(1 + (m-1)\hat{c})}.$$

We reject H_0 if

$$\frac{\hat{\nu}W}{2} > \chi^2_{\hat{\nu};1-\alpha}.$$

Note that it is possible to obtain a negative estimate of correlation. In addition, if $\hat{c} \leq \frac{-1}{m-1}$ then the $Var(W)$ and $\hat{\nu}$ will be negative (undefined). In practice negative values of $\hat{c}$ can be set to either zero or $\left(\frac{-1}{m-1} + \varepsilon\right)$. We provide a further discussion of negative correlation values in the example below.

Example 3: Using the diabetes data presented in Chapter 4, we would like to use combined p-values to detect differentially expressed genes. We will use data for 5 (untreated) diabetic mice and 5 normal mice, each represented on a single array. Each array has 8799 probe sets and most probe sets contain 16 probe pairs. The PM values will be background corrected and normalized according to either the RMA or dChip algorithms. We work on the log_2 scale.

For each probe within each probe set we will test two pairs of hypotheses:

1. $H_0 : \mu_D - \mu_N \geq 0$ versus $H_a : \mu_D - \mu_N < 0$

2. $H_0 : \mu_D - \mu_N \leq 0$ versus $H_a : \mu_D - \mu_N > 0$.

The t-test with Welch-Satterthwaite degrees of freedom will be used. After p-values are computed by probe, the p-values for a probe set will be combined using the method described in this section. We will call this the "combined" p-value. In addition to computing the combined p-value, we also consider what proportion of probes (for probe set) are significant at the 0.05 level. Values of $\hat{c} \leq \frac{-1}{m-1}$ are set to $\frac{-1}{m-1} + \varepsilon$ where $\varepsilon = 0.01$ to prevent negative degrees of freedom.

We will also compute a p-value by probe set based on RMA and dChip expression values using a t-test with Welch-Satterthwaite degrees of freedom. We will use both pairs of hypotheses detailed above. We will call this the "original" p-value.

RMA: A comparison of significant p-values for RMA using original and combined p-values are shown in Tables 8.6 and 8.7. The combined method is able to detect more differentially expressed than the original method.

Table 8.6: Comparison of significant p-values for RMA testing (1) $H_0 : \mu_D - \mu_N \geq 0$ versus $H_a : \mu_D - \mu_N < 0$.

	Combined p-value> 0.05	Combined p-value $\leq$ 0.05
Original p-value> 0.05	6655	1449
Original p-value$\leq$ 0.05	77	618

Table 8.7: Comparison of significant p-values for RMA testing (2) $H_0 : \mu_D - \mu_N \leq 0$ versus $H_a : \mu_D - \mu_N > 0$.

	Combined p-value> 0.05	Combined p-value $\leq$ 0.05
Original p-value> 0.05	7820	492
Original p-value$\leq$ 0.05	9	478

dChip: For dChip, background/normalized probe values less than or equal to zero were set to 0.01 (to prevent undefined logged values). A comparison of significant p-values for dChip using original and combined p-values are shown in Tables 8.8 and 8.9. Once again, the combined method is detecting more differentially expressed genes than the original method.

Table 8.8: Comparison of significant p-values for dChip testing (1) $H_0 : \mu_D - \mu_N \geq 0$ versus $H_a : \mu_D - \mu_N < 0$.

	Combined p-value> 0.05	Combined p-value $\leq$ 0.05
Original p-value> 0.05	7339	1008
Original p-value$\leq$ 0.05	88	364

For either RMA or dChip preprocessed data, the combined method is detecting more differentially expressed genes than the original method. This may be due to the χ^2 approximation (Equation 8.14). This requires further examination.

Table 8.9: Comparison of significant p-values for dChip testing (2) $H_0 : \mu_D - \mu_N \leq 0$ versus $H_a : \mu_D - \mu_N > 0$.

	Combined p-value> 0.05	Combined p-value $\leq$ 0.05
Original p-value> 0.05	7623	626
Original p-value$\leq$ 0.05	17	533

We also examine how the proportion of significant probes varies with the original p-value. A plot showing the proportion of significant probes versus original p-value for hypothesis 1 is shown in Figure 8.6 and hypothesis 2 is shown in Figure 8.7. Notice that for hypothesis 2, there are a few instances where the dChip original p-value is high, but a high proportion of probes is significant. In these cases the methods seem to be giving contradictory information. An extreme example of this is probe set 4099 (rc-AA875084-at) hypothesis 2, where 13/16 probes were significant but the original p-value is 0.631. For RMA, 13/16 probes were also significant and the original p-value is 0.004. The expression values (by array) for dChip (both normalized and "raw") and RMA are shown in Table 8.10. Note that the normalization does not appear to be the explanation for the strange behavior of this probe set.

Table 8.10: P-values and expression values for probe set 4099.

		DM					NM				
Method	p	3.3	3.4	3.5	3.7	6.41	3.1	5.36	5.37	6.49	6.50
dChip	0.631	10.88	11.07	10.89	11.04	10.99	10.80	10.73	11.10	11.36	11.08
dChip(raw)	0.701	10.60	10.30	10.00	11.40	10.60	10.80	10.90	8.47	15.00	10.80
RMA	0.004	8.52	8.10	8.18	8.50	8.26	7.82	6.29	7.48	7.05	6.66

Figure 8.6: Proportion of probes (within a probe set) with p-values less than 0.05 versus original p-value for hypothesis 1.

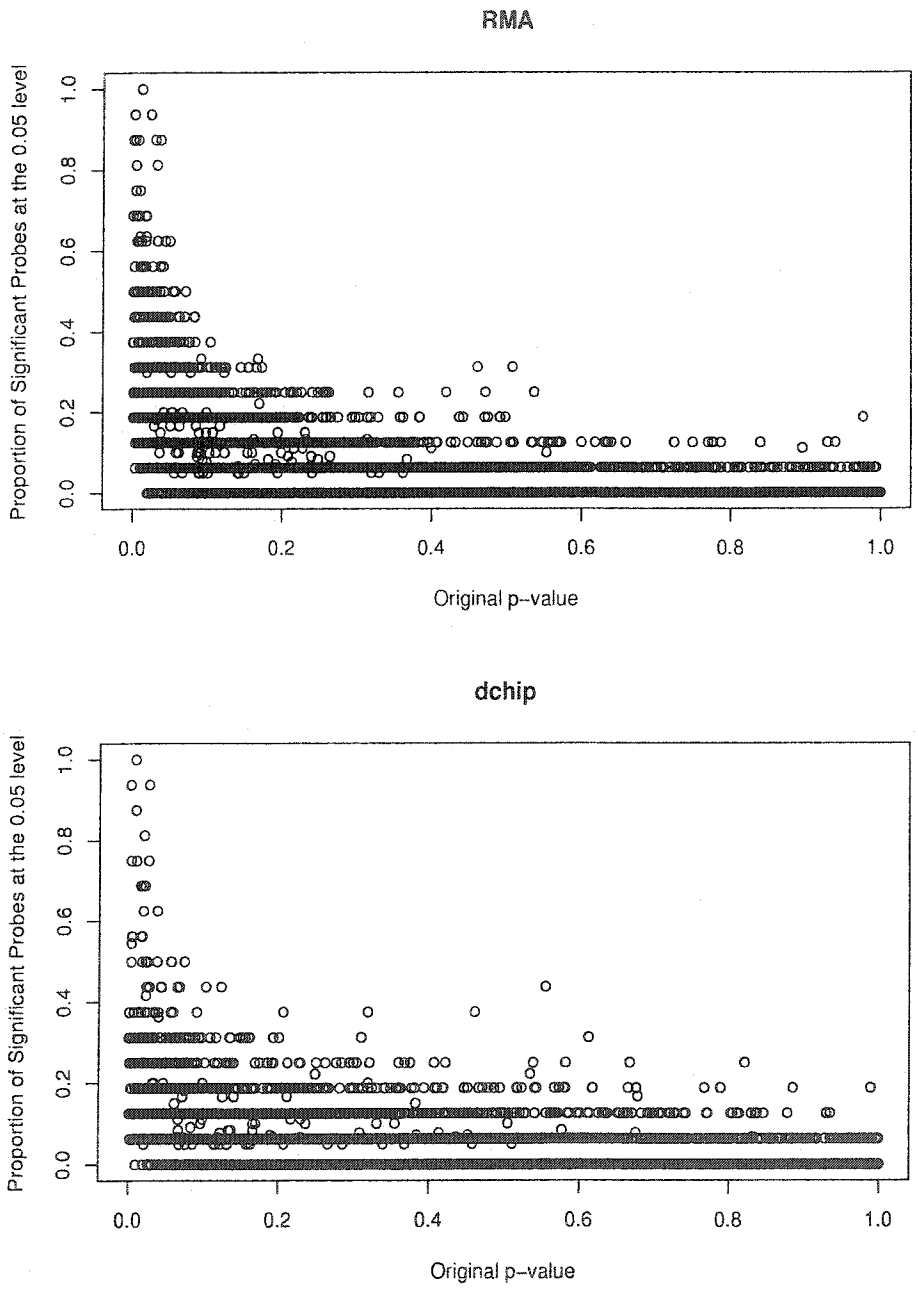
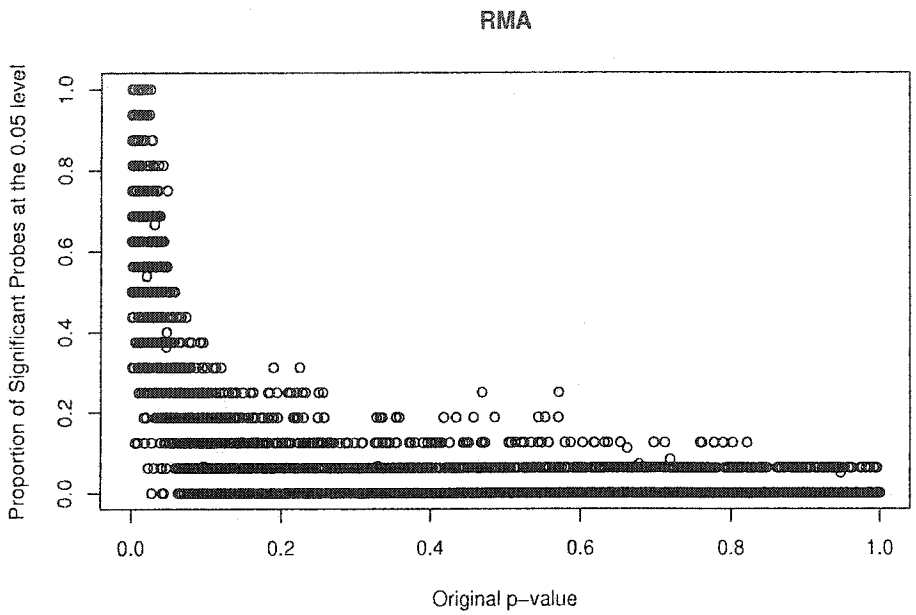
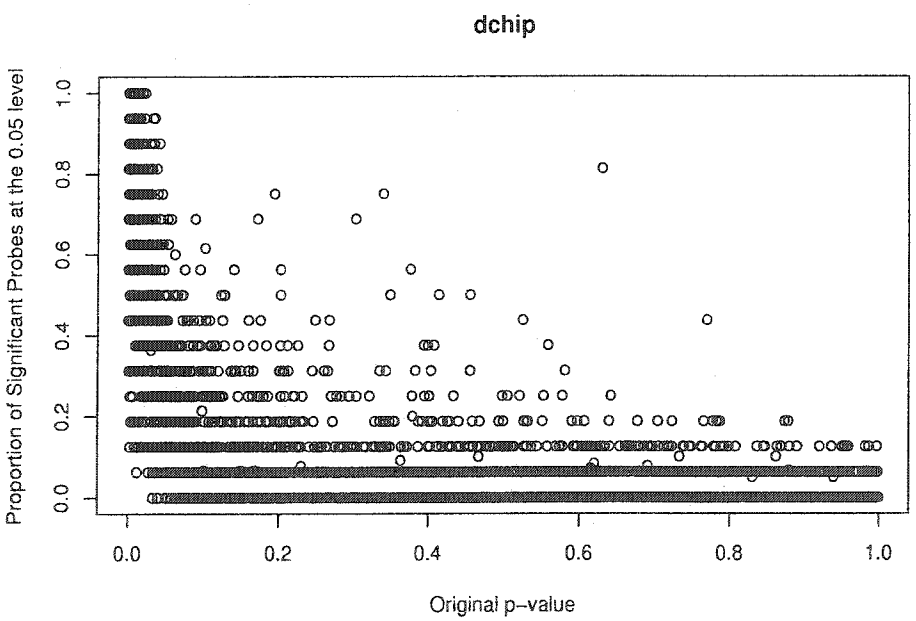


Figure 8.7: Proportion of probes (within a probe set) with p-values less than 0.05 versus original p-value for hypothesis 2.





Negative Estimates of Correlation: For RMA hypothesis 1, 22% of probe sets had negative $\hat{c}$ values and for these probe sets 90% were significant at the 0.05 level based on the combined p-value while only 26% were significant based on the original p-value. For hypothesis 2, 11% of probe sets had negative $\hat{c}$ values and for these probe sets 80% were significant at the 0.05 level based on the combined p-values while 37% were significant based on the original p-value.

For dChip hypothesis 1, 18% of probe sets had negative $\hat{c}$ values and for these probe sets 82% were significant at the 0.05 level based on the combined p-value while 22% were significant based on the original p-value. For hypothesis 2, 13% of probe sets had negative $\hat{c}$ values and for these probe sets 80% were significant at the 0.05 level based on the combined p-value while 33% were significant based on the original p-value. This might indicate that the combined method is more likely than the original method to declare a gene differentially expressed when the estimated correlation for that probe set is negative.

What is the meaning of an estimated negative correlation? Since a positive correlation implies that the probes (within a probe set) have related responses, then a negative correlation implies that probes within a probe set are giving conflicting information. An extreme example of this is Gene 2917 (M69055-at), which has negative $\hat{c}$ values for both RMA and dChip for both sets of hypotheses. When we consider the proportion of probes declared significant, we find that for RMA 3/16 probes had significant p-values for hypothesis 1 and 9/16 probes had significant p-value for hypothesis 2. Similarly for dChip, we find that 3/16 probes had significant p-values for hypothesis 1 and 11/16 probes had significant p-values for hypothesis 2. So, in this case probes appear to be giving conflicting information! Of the 2816 RMA probe sets which had a negative $\hat{c}$ value for either hypothesis 1 or hypothesis 2, only 260 of these probe sets had at least one significant probe for each set of hypotheses. Similarly of the 2545 dChip probe sets which had a negative $\hat{c}$ value for

either hypothesis 1 or hypothesis 2, only 360 of these probe sets had at least one significant probe for each set of hypotheses.

8.4 Recommended Diagnostic Analysis

In this chapter we have proposed a number of probe level diagnostics. Here we summarize the proposed analyses.

1. Preprocess the data.

2. Plot the probe level fold change estimates. The mode of the empirical distribution should be close to one (on the original scale) or zero (on the log scale). If not, consider performing a different or additional normalization.

3. Compute the deviation of the estimated logFC (by probe) compared to the median estimated logFC for the probe set. Obtain the empirical distribution of these deviations over all arrays. Use this empirical distribution to find the 5th and 95th percentiles and use these as bounds for flagging outliers. Outlier probes can be flagged by comparing each pair of arrays. Probe sets that have a consistently large number of outliers (across pairs of arrays) should be inspected further, particularly if the gene (corresponding to such a probe set) is declared differentially expressed.

4. Compute an expression index by gene and array.

5. Compare p-values obtained using expression indices and combined p-values from probe level information using the same significance test for both. Examine those genes which were declared differentially expressed by one method but not the other.

Chapter 9

TRANSCRIPTIONAL REGULATION ANALYSIS USING MICROARRAYS

In order to discuss transcriptional regulation, we must begin with a discussion of transcription factors. A transcription factor is defined as "any protein that is needed for initiation of transcription, but which is not itself part of RNA polymerase" [40, p813]. The promoter region is located upstream of the transcription start site and contains motifs that bind transcription factors. RNA polymerase II (which transcribes mRNA) cannot initiate transcription on its own; it is dependent on transcription factors.

The goal of transcriptional regulation analysis is to identify required transcription factors. This can be accomplished by looking for transcription factors that bind to the promoter region in a group of up or down-regulated genes. A gene whose transcription level is lowered is called down-regulated; a gene whose transcription level is increased is call up-regulated. Rhodius and LaRossa [56] provide a general discussion of transcriptional regulation analysis using microarrays.

Frequently, mutant DNA will be used when studying transcriptional regulation. The mutants are employed to gain insight into normal gene functioning. For example, when wild type (normal) and mutant yeast are compared, we can identify genes that are failing to be transcribed when they have mutant DNA. Of course careful selection of mutants is required: not much information will be gained if *all* genes are down-regulated compared to wild type.

In the search for transcription factors, it might make sense to look at groups of up or down-regulated genes by functional category. It seems reasonable that genes that have the same type of function may be regulated by the same transcription factors. Functional categories for some organisms (including *Saccharomyces cerevisiae* (yeast) and *Arabidopsis Thaliana*) can be obtained from the Munich Information Center for Protein Sequences (MIPS) [45]. Note that a gene may be contained in more than one MIPS category.

Next an over-representation analysis is used to identify MIPS categories that are unusually affected by the mutation. A category is considered to be over-represented in the pool of increasing genes, if that category makes up a larger percentage of the pool of increasing genes than it contributes to the general pool of genes. Specifically, the over-representation ratio (OVR) is used to look for over-representation in increasing genes:

$$OVR = \frac{NI_C/NI}{N_C/N}, \tag{9.1}$$

where NI represents the number of increasing genes, NI_C represents the number of increasing genes in a category, N is the number of genes and N_C is the number of genes in a certain category. Note that an OVR value of 1 indicates that a category is NOT over-represented. The OVR values are calculated by group for both increasing and decreasing genes. Over-represented categories can be identified as outliers in a histogram (or box plot) of OVR values, or by values that exceed a certain threshold.

For over-represented categories, we perform a word search (where a "word" or motif is a specific sequence of bases) on the group of increasing or decreasing genes. We are looking for words that appear more often than expected in the gene sequence. Word searches can be performed using the pattern discovery tool from regulatory sequence analysis (RSA) tools [27]. RSA tools supports a large number of organisms (currently 245). The statistical significance of a motif is "based on tables

of oligonucleotide frequencies observed in all non-coding sequences" [70]. Note that the goal is to find short (5-8 bases in length) highly conserved patterns.

We examine these words to see if they are associated with a transcription factor binding site. PROSPECT maintains a list of known yeast transcription factor binding sites [23]. Transcription factor binding sites for other organisms are available elsewhere. If functional category information is not employed, all up or down-regulated genes can be searched together.

Although we start our search with a functional category, it makes sense that if a transcription factor was actually responsible for up- or down-regulation of genes in that category it would also be associated with up- or down-regulation of any gene for which its binding site appears. Hence for those words that are identified as known transcription factor binding sites, we would like to verify that the effect generalizes beyond the functional category. To do so, we identify all genes that contain the transcription factor binding site and consider what proportion of these genes are up or down-regulated.

The presence of a certain transcription factor binding site does not necessarily mean that the gene is actually regulated by that transcription factor. For example, a gene might have a certain binding site but the location might be too far upstream such that the transcription factor fails to bind. Conversely, just because a gene contains none of the known variants of a certain transcription factor binding site, does not mean that it is not regulated by that transcription factor. A gene might have an unusual variation on a common motif (such that it is not identified by RSA tools), but still be regulated by the transcription factor.

Chapter 10

TRANSCRIPTIONAL REGULATION ANALYSIS: A CASE STUDY

The experiment reported here was conducted to gain insight into TBP's role in transcription in yeast(*Saccharomyces cerevisiae*).

Most promoters have a TATA box (which is the sequence TATAAA) located 25 bp upstream. "It is the only upstream promoter element that has a relatively fixed location with respect to the startpoint. It is found in all eukaryotes" [40, p823]. This fixed location allows for the proper positioning of RNA polymerase II. TFIID is made up of TATA binding protein (TBP) and TBP associated factors. TBP binds to the TATA box and activates transcription. However, TBP does not initiate transcription by itself, but does so with the help of other transcription factors. For example, activators are needed to recruit TBP to the promoter.

Wild type yeast and two TBP mutants (LAS17=F237D and RM5=K151L,K156Y) were considered [65, 54]. The mutants were carefully selected such that TBP was able to bind to the TATA box, but some interactions with other transcription factors may be affected. Since at least one of the mutants was expected to be temperature sensitive, gene expression data was obtained at 30°C and 38°C.

10.1 Materials and Methods

Gene expression data was obtained using GeneChip yeast expression analysis microarrays (Ye6100 set), consisting of a set of four probe arrays that cover the entire yeast genome. For this experiment, wild type (WT) at 30°C or 38°C was considered

baseline and two mutants at 30°C and 38°C were considered the experimental data. Each experimental probe array was analyzed to obtain and absolute call (absent or present). In addition, the data was analyzed using the "comparison algorithms" to identify differences between the experimental and baseline arrays for every gene represented on the array. Fold change values were calculated for each array. The difference call indicates whether a gene has increased (I), decreased (D), or exhibits no change (NC) relative to baseline. Note that no CEL files ("raw" data) were available for this experiment. Hence, the present(P)/absent(A) calls, difference calls and fold change values obtained from MAS 4.0 were used with the following modifications:

- The difference call for any probe set for which the P/A call in the baseline sample was A, the P/A call in the experimental sample was P and the difference call was D was changed to NE (not evaluable).
- Similarly, the difference call for any probe set for which the P/A call in the baseline sample was P, the P/A call in the experimental sample was A, and the difference call was I was changed to NE.
- The difference call for any probe set where the average difference intensity in both baseline and experimental samples was less than 100 was changed to NE (not expressed).
- The difference call for any probe set where fold change was between -2.0 and 2.0 was changed to NC (no change).

These changes reduce the number of difference calls to four. Hence the fold change and difference calls were obtained comparing mutant to WT at two temperatures: F237D versus WT at 30°C, F237D versus WT at 38°C, K151L,K156Y versus WT at 30°C, and K151L,K156Y versus WT at 38°C. We also considered WT at 38°C versus 30°C.

Table 10.1: Comparison of wild type at 38°C versus 30°C

DiffCall for WT at 38°C vs 30°C				
D	I	NC	NE	Total
80	201	4213	1927	6421

10.2 General Comparison

We start with some general summary statistics of the experiment. Recall that a fold change of magnitude 2.0 or greater was required to be classified as increasing or decreasing. In addition, all difference calls are relative to wild type.

A table of difference calls for wild type (WT) at 38°C versus 30°C is shown in Table 10.1. We see that 1.7% of expressed WT genes (all genes except those that are not expressed) are decreasing at 38°C as compared to 30°C. An additional 4.5% of expressed genes are increasing. However, the majority of expressed genes (93.7%) are not changed across temperatures.

A comparison of F237D across temperatures is shown in Table 10.2. We see that at 30°C, 5.1% of expressed genes were classified as decreasing for F237D versus 4.7% at 38°C. At 38°C, 15.3% of expressed genes were classified as increasing for F237D versus 7.6% at 38°C.

A comparison of K151L,K156Y across temperatures is shown in Table 10.3. As expected K151L,K156Y appears to be temperature sensitive [54]. We see that at 30°C, only 2.5% of expressed genes were classified as decreasing for K151L,K156Y as compared to 9.0% at 38°C. At 30°C, 5.6% of expressed genes were classified as increasing for K151L,K156Y versus 11.3% at 38°C.

We also compare the mutants to each other. Table 10.4 gives a comparison of F237D and K151L,K156Y at 30°C. Table 10.5 gives a comparison of the two mutants at 38°C.

Table 10.2: Comparison of F237D at 30°C versus 38°C

	DiffCall for F237D at 38°C				
DiffCall for F237D at 30°C	D	I	NC	NE	Total
D	56	26	146	11	239
I	20	106	495	91	712
NC	120	194	3248	151	3713
NE	24	27	212	1494	1757
Total	220	353	4101	1747	6421

Table 10.3: Comparison of K151L,K151Y at 30°C versus 38°C

	DiffCall for K151L,K156Y at 38°C				
DiffCall for K151L,K156Y at 30°C	D	I	NC	NE	Total
D	75	2	27	7	111
I	3	77	153	14	247
NC	303	384	3278	117	4082
NE	44	75	321	1541	1981
Total	425	538	3779	1679	6421

Table 10.4: Comparison of F237D versus K151L,K156Y at 30°C

	DiffCall for K151L,K156Y at 30°C				
DiffCall for F237D at 30°C	D	I	NC	NE	Total
D	30	3	206	0	239
I	19	119	409	165	712
NC	62	91	3380	180	3713
NE	0	34	87	1636	1757
Total	111	247	4082	1981	6421

Table 10.5: Comparison of F237D versus K151L,K156Y at 38°C

	DiffCall for K151L,K156Y at 38°C				
DiffCall for F237D at 38°C	D	I	NC	NE	Total
D	89	3	128	0	220
I	21	107	172	53	353
NC	315	351	3324	111	4101
NE	0	77	155	1515	1747
Total	425	538	3779	1679	6421

10.3 Transcriptional Regulation Analysis by MIPS Functional Category

A transcriptional regulation analysis was performed using the algorithm outlined in Chapter 9. First, genes were divided into MIPS functional categories. Then over-represented categories were identified based on OVR values. Here, an OVR value of six or greater was considered over-represented. A list of over-represented categories (with an OVR value of six or greater) for F237D are shown in Table 10.6 and those categories over-represented for K151L,K156Y are shown in Table 10.7.

Table 10.6: Over-represented categories for F237D with the proportion of genes in the category that are identified as increasing or decreasing and the OVR value.

MIPS Category	proportion	Temp	I/D	OVR
ageing	1/3	38	I	6.06
amino acid transporters	5/22	30	D	6.11
allantoin/allantoate tranporters	2/9	38	D	6.49
amino acid metabolism	44/195	38	D	6.59
nitrogen and sulphur metabolism	17/73	38	D	6.80
amino acid metabolism	52/195	30	D	7.16
glyoxylate cycle	5/6	30	I	7.52
other protein destination activities	2/6	30	D	8.96
amino acid transporters	7/22	38	D	9.29

Table 10.7: Over-represented categories for K151L,K156Y with the proportion of genes in the category that are identified as increasing or decreasing and the OVR value.

MIPS Category	proportion	Temp	I/D	OVR
amino acid transporters	10/22	38	D	6.87
pentose-phosphate pathway	4/8	38	D	7.55
glyoxylate cycle	2/6	30	I	8.67
extracellular/secretion proteins	3/20	30	D	8.68
biogenesis(cell membrane)	1/1	38	I	11.93
biogenesis(intracell transport vesicles)	1/1	38	I	11.93
allantoin and allantoate tranporters	3/9	30	D	19.28

Table 10.8: Amino acid metabolism (F237D decreasing at 30°C)

	Decrease		No Change			Transcription
	N	Proportion	N	Proportion	Difference	Factor
GAGTCA	29	0.558	43	0.384	0.174	GCN4 (GAGTCA)
AGTCAT	32	0.615	44	0.393	0.223	
GACTCA	27	0.519	29	0.259	0.26	
GCCACA	24	0.462	20	0.179	0.283	
ACTGTG	19	0.365	26	0.232	0.133	ADR1(...ACTGTG...)
Overall	52		112			

Table 10.9: Amino acid metabolism (F237D decreasing at 38°C)

	Decrease		No Change			Transcription
	N	Proportion	N	Proportion	Difference	Factor
GAGTCA	26	0.591	50	0.388	0.203	GCN4 (GAGTCA)
AGTCAT	27	0.614	52	0.403	0.211	
GCCACA	23	0.523	25	0.194	0.329	
CCACAG	21	0.477	26	0.202	0.276	ADR1 (...CCACAG...)
ACTGTG	21	0.477	27	0.209	0.268	ADR1 (...ACTGTG...)
Overall	44		129			

Only MIPS categories with 10 or more (increasing or decreasing) genes were considered in further analysis. This allows for meaningful word searches. The included categories were: amino acid metabolism (F237D decreasing at 30°C and 38°C), nitrogen and sulfur metabolism (F237D decreasing at 38°C) and amino acid transporters (K151L,K156Y decreasing at 38°C).

For each of these categories, a word search was performed on the decreasing (or increasing) subset of genes using the pattern discovery tool form the RSA tools website [27]. A list of "significant" six letter words (with a high number of occurrences) was compiled. The summaries of each category for these "most significant" words (by MIPS category) are given in Tables 10.8 through 10.11. Each of the words listed were examined to see if they could be matched with a transcription factor binding site using PROSPECT [23]. The matches are also shown in Tables 10.8 through 10.11.

Table 10.10: Nitrogen and sulfur metabolism (F237D decreasing at 38°C)

	Decrease		No Change			Transcription
	N	Proportion	N	Proportion	Difference	Factor
CACGTG	10	0.588	4	0.095	0.493	CPF1 (CACGTGA)
ACGTGA	13	0.765	15	0.357	0.408	CPF1 (CACGTGA)
CGTGAC	9	0.529	6	0.143	0.387	
AGTCAT	11	0.647	16	0.381	0.266	
Overall	17		42			

Table 10.11: Acid Transporters (K151L,K156Y decreasing at 38°C).

	Decrease		No Change			Transcription
	N	Proportion	N	Proportion	Difference	Factor
CCACAG	8	0.800	2	0.222	0.578	ADR1 (..CCCACAG...)
GCCACA	8	0.800	1	0.111	0.689	
ACTGTG	7	0.700	4	0.444	0.256	ADR1 (...ACTGTG...)
CGGCGC	4	0.400	3	0.333	0.067	
CGCCAA	7	0.700	3	0.333	0.367	CAR1 repressor (...CGCCAA)
CGCCAC	7	0.700	0	0	0.700	
CAGTTC	8	0.800	2	0.222	0.578	
Overall	10		9			

Note that the ADR1 and RAP1 binding sites are quite long. In order to take a closer look at these "matches", another word search was performed, this time allowing two substitutions (the maximum allowed by RSA tools). Using this criteria, ADR1 was not identified in any of the amino acid metabolism genes (at either temperature) or any of the amino acid transporter genes. RAP1 was only located in a single gene from the metabolism of energy reserves group. Due to its length, no substitutions were allowed for the repressor of CAR1. It was not identified in any of the amino acid transporter genes.

10.4 A Possible Link Between GCN4 and TBP

GCN4 (SGD name YEL009C, alias AAS3 or ARG9) encodes a transcriptional activator. It is well known that GCN4p stimulates transcription of amino acid biosynthetic genes in response to starvation for any of several amino acids [29].

Table 10.12: Difference calls for F237D genes at 30°C with and without GCN4 binding sites (as given by PROSPECT), p=0.162.

	GCN4 Binding Site		
F237D at 30 C	No	Yes	Total
D	196	40	236
I	584	108	692
NC	2889	659	3548
NE	1481	189	1670
Total	5150	996	6146

Table 10.13: Difference calls for F237D genes at 38°C with and without GCN4 binding sites (as given by PROSPECT), p=0.000.

	GCN4 Binding Site		
F237D at 38°C	No	Yes	Total
D	197	18	215
I	297	51	348
NC	3188	736	3924
NE	1468	191	1659
Total	5150	996	6146

There is also evidence that GCN4p is induced under conditions of stress besides amino acid starvation [29]. For F237D at 30°C and 38°C, YEL009C was classified as no change.

There is a tendency for F237D amino acid metabolism genes to be down regulated when they contain GAGTCA binding site for GCN4. A search performed using PROSPECT revealed that there are 1000 genes that contain an upstream binding site for GCN4. The results for all genes at 30°C are shown in Table 10.12. The results for all genes at 38°C are shown in Table 10.13. The p-value for the χ^2 test for the D, I, and NC rows of the tables are also given.

Note that at 30°C, 17% of decreasing genes contain a GCN4 binding site versus 19% of no change genes. At 38°C, 8% of decreasing genes contain a GCN4 binding

Table 10.14: Difference calls for F237D genes at 30°C identified as GCN4 target or non-target genes, p=0.000.

F237D at 30°C	GCN4 Target No	Yes	Total
D	168	68	236
I	655	37	692
NC	3259	289	3548
NE	1555	115	1670
Total	5637	509	6146

Table 10.15: Difference calls for F237D genes at 38°C identified as GCN4 target or non-target genes, p=0.000.

F237D at 38°C	GCN4 Target No	Yes	Total
D	156	59	215
I	307	41	348
NC	3628	296	3924
NE	1546	113	1659
Total	5637	509	6146

site versus 19% of no change Genes. Thus the results that we saw when considering only amino acid metabolism genes have been greatly obscured.

Natarajan *et al.* identified 539 genes that were classified as GCN4 targets (the list supplied by Hinnebusch contained 512 genes) [49]. Some of these GCN4 targets contain known GCN4 binding sites and others do not. The results for all genes at 30°C are shown in Table 10.14. The results for all genes at 38°C are shown in Table 10.15. The p-value for the χ^2 test for the D, I, and NC rows of the tables are also given.

Looking at GCN4 targets (instead of genes with GCN4 binding sites) we see results similar to those obtained when we considered only the amino acid metabolism genes. At 30°C, 29% of decreasing genes were GCN4 targets versus 5% of increasing

genes and 8% of no change genes. At 38°C, 27% of decreasing genes were GCN4 targets versus 12% of increasing genes and 8% of no change genes.

Note that at 30°C, 17% of the expressed GCN4 target genes were down-regulated versus 4% of non-target genes (p=0.000). At 38°C, 15% of expressed GCN4 target genes were down-regulated versus 4% of non-target genes (p=0.000).

A complete listing of 67 amino acid metabolism genes which are decreasing for F237D at either 30 C or 38 C is given in Tables 10.16 and 10.17. Note that some genes are GCN4 targets but do not contain a GCN4 binding site (defined here as GAGTCA). On the other hand, some genes contain GCN4 binding sites but are not GCN4 targets. Also note that some genes with "large" fold change values (i.e. YJR010w) cannot be explained by either a GCN4 target or a binding site.

10.5 A Possible Link Between CPF1 and TBP

CPF1 (SGD name YJR060W, alias CBF1 or CEP1) is centromere binding factor, induces DNA bending and is required for mitotic segregation and normal growth rate. For F237D at 38°C, YJR060W was classified as no change.

There is a tendency for F237D nitrogen and sulfur metabolism genes to be down regulated when they contain a CPF1 binding site. A search performed using PROSPECT revealed that there are 531 genes that contain an upstream binding site for CPF1. The results for all genes are shown in Table 10.18. The p-value for the χ^2 test for the D, I, and NC rows of the table is also given.

Hence at 30 C, 14% of decreasing genes contained a CPF1 binding site versus 8% of increasing genes and 9% of no change genes. Also, 7% of expressed genes with CPF1 binding sites were down-regulated versus 5% of expressed genes without such binding sites (p=0.036).

A complete list of Nitrogen and sulfur metabolism genes with are decreasing for F237D at 38°C is given in Table 10.19. Note that two of the genes with the largest negative fold changes do not contain a binding site for CPF1.

Table 10.16: Amino acid metabolism genes which are decreasing for F237D.

SGD Name	Gene	GCN4 Target	GCN4 BS	F237D at 30°C FC	F237D at 30°C DiffCall	F237D at 38° FC	F237D at 38° DiffCall
YBR006w	UGA2	Yes	No	-3.6	D	2	NC
YBR084w	MIS1	No	No	-1.5	NC	-2.1	D
YBR115c	LYS2	Yes	Yes	-2	D	-2.6	D
YBR213w	MET8	No	Yes	-3.3	D	-5.1	D
YBR248c	HIS7	Yes	Yes	-2.6	D	-3.2	D
YBR253w	SRB6	No	No	-4	D	-1.2	NC
YBR294w	SUL1	No	Yes	-8.7	D	-22.5	D
YCL009c	ILV6	Yes	Yes	-1.6	NC	-2	D
YCL025c	AGP1	Yes	Yes	-1	NC	-3.3	D
YCL030c	HIS4	Yes	Yes	-3.1	D	-2.5	D
YDL048c	STP4	No	No	-2.3	D	1.3	NC
YDL171c	GLT1	Yes	Yes	-2	D	-1.5	NC
YDL215c	GDH2	No	Yes	1.7	NC	-2.4	D
YDR046c	BAP3	No	No	-2.6	D	-1.9	NC
YDR158w	HOM2	Yes	Yes	-2.7	D	-1.4	NC
YDR253c	MET32	No	No	-5.7	D	-4.1	D
YDR502c	SAM2	No	No	-4.1	D	-1.7	NC
YER023w	PRO3	No	Yes	-2	D	1.2	NC
YER042w	MXR1	No	No	-2.5	D	-2.8	D
YER052c	HOM3	Yes	Yes	-1.6	NC	-2	D
YER069w	ARG5,6	Yes	Yes	-2.7	D	-3.2	D
YER081w	SER3	Yes	No	1.3	NC	-3.9	D
YER090w	TRP2	Yes	Yes	-2.1	D	-1.5	NC
YER091c	MET6	No	Yes	-9.8	D	-2.7	D
YFL018c	LPD1	Yes	Yes	-2.5	D	-1.4	NC
YFL055w	AGP3	No	No	-2	NC	-3.4	D
YFR030w	MET10	Yes	No	-6.9	D	-6.3	D
YGL009c	LEU1	Yes	Yes	-2	D	1.1	NC
YGL125w	MET13	Yes	Yes	-4.4	D	-2.4	D
YGL184c	STR3	Yes	Yes	-7.2	D	-5	D
YGR055w	MUP1	No	Yes	-3.3	D	-2.3	D
YGR208w	SER2	No	Yes	-2.5	D	-1.9	NC
YHL036w	MUP3	Yes	Yes	-4.4	D	-2.5	D
YHR018c	ARG4	Yes	Yes	-2.3	D	-2.1	D
YHR208w	BAT1	Yes	Yes	-2	D	-1.1	NC
YIL046w	MET30	No	No	1.6	NC	-2.3	D
YIL074c	SER33	Yes	Yes	-4	D	-3.8	D
YIL094c	LYS12	Yes	Yes	-2.2	D	1	NC
YIL116w	HIS5	Yes	Yes	-2.6	D	-2.6	D

Table 10.17: More amino acid metabolism genes which are decreasing for F237D.

SGD Name	Gene	GCN4 Target	GCN4 BS	F237D at 30°C FC	F237D at 30°C DiffCall	F237D at 38° FC	F237D at 38° DiffCall
YIR017c	MET28	Yes	Yes	-20.6	D	-5.2	D
YIR034c	LYS1	Yes	Yes	-3.8	D	-1.7	NC
YJR010w	MET3	No	No	-7.9	D	-20.9	D
YJR025c	BNA1	Yes	Yes	-8.2	D	-1.4	NC
YJR078w	YJR078w	No	Yes	-1	NC	-2.2	D
YJR109c	CPA2	Yes	Yes	-1.4	NC	-2.4	D
YJR130c	STR2	Yes	No	1.1	NC	-2.5	D
YJR137c	ECM17	Yes	Yes	-7.3	D	-11	D
YJR139c	HOM6	No	No	-3.7	D	1.1	NC
YKL001c	MET14	No	No	-4.9	D	-2.3	D
YKL211c	TRP3	Yes	Yes	-2.2	D	1	NC
YKL218c	SRY1	Yes	Yes	-4.4	D	-5.9	D
YLL061w	MMP1	No	No	-1.7	NC	-4.3	D
YLR092w	SUL2	Yes	No	-8.3	D	-15.5	D
YLR303w	MET17	Yes	No	-5.3	D	-1.7	NC
YMR062c	ECM40	Yes	Yes	-3.2	D	-2.7	D
YMR108w	ILV2	Yes	No	-3.2	D	-1.2	NC
YNL277w	MET2	Yes	No	-8.8	D	-6	D
YNR050c	LYS9	Yes	No	-3.1	D	1	NC
YOL058w	ARG1	Yes	Yes	-1.5	NC	-2.2	D
YOL064c	MET22	Yes	Yes	-2.8	D	-1.4	NC
YOR130c	ORT1	Yes	Yes	-2.2	D	-2	D
YOR184w	SER1	Yes	Yes	-3.1	D	-1.7	NC
YOR202w	HIS3	Yes	Yes	-1.6	NC	-2.2	D
YOR375c	GDH1	No	Yes	-3.2	D	-1.6	NC
YPL274w	SAM3	No	No	-11.8	D	-3.8	D
YPR035w	GLN1	No	Yes	-1.4	NC	-2.4	D
YPR167c	MET16	Yes	Yes	-3.8	D	-5.9	D

Table 10.18: Difference calls for F237D genes at 38°C with and without CPF1 binding sites, p=0.035.

F237D at 38°C	CPF1 Binding Site No	Yes	Total
D	184	31	215
I	319	29	348
NC	3557	367	3924
NE	1559	100	1659
Total	5619	527	6146

Table 10.19: Nitrogen and sulfur metabolism genes that are decreasing for F237D at 38°C.

SGD Name	Gene	Fold Change	CPF1 Binding Site
YBR294w	SUL1	-22.5	No
YJR010w	MET3	-20.9	Yes
YLR092w	SUL2	-15.5	Yes
YAL067c	SEO1	-15.1	No
YJR137c	ECM17	-11	Yes
YFR030w	MET10	-6.3	Yes
YPR167c	MET16	-5.9	Yes
YIR017c	MET28	-5.2	Yes
YBR213w	MET8	-5.1	Yes
YDR253c	MET32	-4.1	Yes
YDR242w	AMD2	-3	No
YDL215c	GDH2	-2.4	No
YJL060w	YJL060w	-2.4	Yes
YPR035w	GLN1	-2.4	No
YDL170w	UGA3	-2.3	No
YKL001c	MET14	-2.3	Yes
YOL058w	ARG1	-2.2	No

Table 10.20: Count of genes with CPF1 BS (Binding Site) or GCN4 targets (F237D at 38°C)

CPF1 Binding Site	No	Yes	No	Yes	
GCN4 Target	No	No	Yes	Yes	
LAS17 at 38°C					Total
D	138	18	46	13	215
I	283	24	36	5	348
NC	3301	327	256	40	3924
NE	1450	96	109	4	1659
Total	5172	465	447	62	6146

It is interesting to see if we can explain more of the decreasing difference calls for F237D at 38°C, by combining results for CPF1 binding sites and GCN4 target genes. These results are summarized in Table 10.20.

We see that 36% of decreasing genes are GCN4 targets or contain a CPF1 binding site or both versus 19% of increasing genes and 16% of no change genes.

10.6 Summary of Results

Two mutant strains of yeast (F237D and K151L,K156Y) that produce altered TBP were studied at 30°C and 38°C using microarray technology. Using MIPS categories, an over-representation analysis was performed. After identifying four MIPS categories that were over-represented (by contributing more increasing or decreasing genes than expected), a word search was performed for each of the four MIPS categories. Then the results of the word search were examined to see if they corresponded to a transcription factor binding site. The binding sites for GCN4 and CPF1 were identified in this manner. Genes that were regulated by GCN4 and/or CPF1 had a tendency to be down-regulated. This indicates that for these genes, GCN4 and/or CPF1 is required to activate transcription.

Chapter 11

CONCLUSIONS AND FUTURE WORK

11.1 Conclusions

In order to summarize the conclusions of this dissertation, we step through a typical microarray experiment.

Before a microarray experiment is even conducted, the experimenter must decide how many arrays to use. In Chapter 5, we discussed the use of SimArray as a sample size calculator. The required input includes one or two "starter" arrays (possibly taken from a previous experiment), a list of proposed fold changes and variance components estimates. The user must also choose a stated model (RMA or MBEI). From this initial input, SimArray simulates microarray data for a requested number of replicates from which the power and false discovery rate can be estimated. This provides the experimenter a basis for choosing the number of arrays to use in the proposed experiment.

After raw data (in the form of CEL files for Affymetrix arrays) is obtained for a microarray experiment, preprocessing is performed as a first step in analyzing this data. This preprocessing includes optical background correction, normalization and possibly nonspecific binding correction. In Chapter 6, we reviewed commonly used normalization techniques and provide illustrative examples of some strengths and weaknesses of these methods. We also presented a simulation study comparing quantile, invariant set and scale normalization algorithms. Based on this simulation study and rationale behind the methods, we advocate the use of invariant set normalization. In Chapter 8, we proposed some probe level diagnostics which can be

used to check the preprocessing of the data. We also present a method for combining probe level tests of differentially expression.

After preprocessing, a model is fit to the data. In Chapter 3, we outline commonly used models for oligo arrays. We compare the performance of RMA, MBEI and MAS5.0 using SimArray (programmed in Bioconductor) in Chapter 4. Unlike a spike-in or dilution experiments, a simulation study allows for manipulation of many sources of variation. Based on our simulation study, MBEI (with invariant set normalization) is recommended because it maintains its stated FDR while operating with high power.

As an alternative to the standard preprocessing and modelling steps, we propose a unified model for microarrays. The benefit of a unified model is that all preprocessing is combined into a single model.

After preprocessing and modelling of the data, fold change estimates can be obtained. After FC estimation, some additional analysis is almost always performed. One goal of microarray experiment might be to identify required transcription factors. We discussed the process of identifying possible transcription factors in Chapters 9 and 10.

Other goals of microarray experiments include identification of gene function, class discovery and class prediction. Even more generally, microarray experiments are performed to examine the expression profile under a certain condition and compare profiles across conditions. For these goals, clustering and classification algorithms are used. These methods are usually applied to a group of differentially expressed genes. Unsupervised clustering methods which have been applied to microarray data include hierarchical clustering [20], K-means clustering [36] and self-organizing maps [67]. Supervised clustering methods used for microarray data include support vector machines [12] and gene shaving [26].

11.2 Future Work

Currently, the selection of probes, probe locations, house-keeping genes, and the use of replicates are chosen by manufacturers (Affymetrix or Agilent). It would be interesting to explore the opportunities and benefits of statistical design concepts for microarray chip design.

A closer examination of NSB estimation is required. Estimation of NSB is an extremely important step in order to find accurate estimates of FC. Recent work ([76],[48]) indicates that NSB affinity estimates can found based only on the sequence of the oligo. The reliability of these estimates has not been adequately established, but will certainly impact FC estimation.

Recent developments regarding estimation of NSB should be incorporated into SimArray. After these modifications are made, the performance of GC-RMA should be examined using the SimArray framework. SimArray can be used to compare normalization and analysis methods under other conditions. Hopefully, this would further clarify performance differences between the methods.

For the proposed unified model, it would be interesting to compare the fold change estimates obtained by assuming that NSB is the same for PM and MM members of a probe pair versus the estimates obtained by assuming that GSB for MM is zero. Maximum likelihood estimation and other estimation techniques can be applied to the unified model. The performance of the unified model can be compared to other commonly used methods. Diagnostics can be created specifically for the assumptions needed to estimate FC from the unified model.

References

[1] Affymetrix corporate fact sheet. www.affymetrix.com.

[2] Affymetrix. Statistical algorithms reference guide. Affymetrix, Santa Clara, CA, 2001.

[3] Affymetrix. Statistical algorithms description document. Affymetrix, Santa Clara, CA, 2002.

[4] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J.D. Watson. *Molecular Biology of The Cell.* Garland Publishing, New York, NY, 1983.

[5] M. Bakay, Y. Chen, R. Borup, P. Zhao, K. Nagaraju, and E. P. Hoffman. Sources of variability and effect of experimental approach on expression profiling data interpretation. *BMC Bioinformatics*, 3, 2002.

[6] P. Baldi and A.D. Long. A bayesian framework for the analysis of microarray expression data: regularized t-test and statistical inferences of gene changes. *Bioinformatics*, 17:509–519, 2001.

[7] Y. Barash, E. Dehan, M. Krupsky, W. Franklin, M. Geraci, N. Friedman, and N. Kaminski. Comparative analysis of algorithms for signal quantitation from oligonucleotide microarrays. *Bioinformatics*, 20(6):839–846, 2004.

[8] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society Series B*, 57:289–300, 1995.

[9] M.A. Black and R.W.Doerge. Calculation of the minimum number of replicate spots required for detection of significant gene expression fold change in microarray experiments. *Bioinformatics*, 18(12):1609–1612, 2002.

[10] B. Bolstad. affy: Built-in processing methods. BioConductor Vignette, February 2005.

[11] B.M. Bolstad, R.A. Irizarry, M. Astrand, and T.P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19:185–193, 2003.

[12] M.P.S. Brown, W.N. Grundy, D. Lin, N. Cristianini, C.W. Sugnet, T.S. Furey, Jr. M. Ares, and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences*, 97(1):262–267, 2000.

[13] S.E. Choe, M. Boutros, A.M. Michelson, G.M. Church, and M.S. Halfon. Preferred analysis methods for affymetrix genechip revealed by a wholly defined control dataset. *Genome Biology*, 6(R16), 2005.

[14] T.-M. Chu, B. Weir, and R. Wolfinger. A systematic statistical linear modeling approach to oligonucleotide array experiments. *Mathematical Biosciences*, 176:35–51, 2002.

[15] E. Chudin, R. Walker, A. Kosaka, S.X. Uw, D. Rabert, T.K. Chang, and D.E. Kreder. Assessment of the relationship between signal intensities and transcript concentration for affymetrix genechip arrays. *Genome Biology*, 3, 2001.

[16] L.M. Cope, R.A. Irizarry, H.A. Jaffee, Z. Wu, and T.P. Speed. A benchmark for affymetrix genechip expression measures. *Bioinformatics*, 20(3):323–332, 2004.

[17] S. Dudoit, Y.H. Yang, M.J. Callow, and T.P. Speed. Statistical methods for identifying differentially expressed genes in replicated cdna microarray experiments. *Stanford University Technical Report 578*, 2000.

[18] B.P. Durbin, J.S. Hardin, D.M. Hawkins, and D.M. Rocke. A variance-stabilizing transformation for gene-expression microarray data. *Bioinformatics*, 18:S105–S110, 2002.

[19] B. Efron, R. Tibshirani, J.D. Storey, and V. Tusher. Empirical bayes analysis of a microarray experiment. *JASA*, 96:1151–1160, 2001.

[20] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95:14863=14868, 2000.

[21] A. Emmett. The state of bioinformatics. *The Scientist*, 14, 2000.

[22] R.A. Fisher. *Statistical Methods for Research Workers*. Oliver and Boyd, London, 4th edition edition, 1932.

[23] W. Fujibuchi, J.S. Anderson, and D. Landsman. Prospect improves cis-acting regulatory element prediction by integrating expression profile data with consensus pattern searches. *Nucleic Acids Research*, 29:3988–96, 2001. http://seq.cbrc.jp/ wataru/PROSPECT/.

[24] S.C. Geller, J.P. Gregg, P. Hagerman, and D.M. Rocke. Transformation and normalization of oligonucleotide microarray data. *Bioinformatics*, 19(14):1817–1823, 2003.

[25] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Bloomfield, and E.S. Lander. Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–536, 1999.

[26] T. Hastie, R. Tibshirani, M.B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W.C. Chan, D. Botstein, and P. Brown. 'gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology*, 1(2):research0003.1–0003.21, 2000.

[27] J.V. Helden, B. Andre, and J. Collado-Vides. A web site for the computational analysis of yeast regulatory sequences. *Yeast*, 16:177–187, 2000. http://rsat.ulb.ac.be/rsat/.

[28] A.A. Hill, E.L. Brown, M.Z. Whitley, G. Tucker-Kellogg, C.P. Hunter, and D.K. Slonim. Evaluation of normalization procedures for oligonucleotide array data based on spiked crna controls. *Genome Biology*, 2:research0055.1–0055.13, 2001.

[29] A.G. Hinnebusch and K. Natarajan. Gcn4p, a master regulator of gene expression, is controlled at multiple levels by diverse signals of starvation and stress. *Eukaryotic Cell*, 1:22–32, 2002.

[30] R. Hoffmann, T. Seidl, and M. Dugas. Profound effect of normalization on detection of differentially expressed genes in oligonucleotide microarray data analysis. *Genome Biology*, 3(7):research0033.1–0033.11, 2002.

[31] D. Hwang, W.A. Schmitt, G. Stephanopoulos, and G. Stephanopoulos. Determination of minimum sample size and discriminatory expression patterns in microarray data. *Bioinformatics*, 18(9):1184–1193, 2002.

[32] R.A. Irizarry, B.M. Bolstad, F. Collin, L.M. Cope, B. Hobbs, and T.P. Speed. Summaries of affymetrix genechip prove level data. *Nucleic Acids Research*, 31(4 e15), 2003.

[33] R.A. Irizarry, B. Hobbs, F. Collin, Y.D. Beazer-Barclay, K.J. Antonellis, U. Scherf, and T.P. Speed. Exploration, normalization, and summaries of high density oligonucleotide array probe level data. *Biostatistics*, 4(2):249–264, 2003.

[34] R.A. Irizarry, D. Warren, R. Spencer, S. Biswal, B.C. Frank, E. Gabrielson, J.G.N. Garci, J. Geoghegan, G. Germino, C. Griffin, S.C. Hilmer, E. Hoffman, A.E. Jedlicka, E. Kawasaki, I.F. Kim, L. Morsberger, H. Lee, D. Peterson, J. Quackenbush, A. Scott, M. Wilson, Y. Yang, S.Q. Ye, and W. Yu. Multiple labe comparison of microarray platforms. Technical Report Paper 71, Johns Hopkins University, 2004.

[35] T.B. Kepler, L. Crosby, and K.T. Morgan. Normalization and analysis of dna microarray data by self-consistency and local regression. *Genome Biology*, 3(7):research0037.1–research0037–12, 2002.

[36] S. Knudsen. *A Biologist's Guide to Analysis of DNA Microarray Data.* John Wiley and Sons, New York, NY, 2002.

[37] W.P. Kuo, T.K. Jenssen, A.J. Butte, L. Ohno-Machado, and Isaac S. Kohane. Analysis of matched mrna measurements from two different microarray technologies. *Bioinformatics*, 18:405–412, 2002.

[38] E.S. Lander. Array of hope. *Nature Genetics Supplement*, 21:3–5, 1999.

[39] M.-L.T Lee and G.A. Whitmore. Power and sample size for dna microarray studies. *Statistics in Medicine*, 21:3543–3570, 2002.

[40] B. Lewin. *Genes VI.* Oxford University Press, Oxford, England, 1997.

[41] C. Li and W.H. Wong. Model-based analysis of oligonucleotide arrays: Expression index computation and outlier detection. *PNAS*, 98:31–36, 2001.

[42] C. Li and W.H. Wong. Model-based analysis of oligonucleotide arrays: model validataion, design issues and standard error application. *Genome Biology*, 2:research0032.1–0032.11, 2001.

[43] D.J. Lockhart, H. Dong, M.C. Byrne, M.T. Follettie, M.V. Gallo, M.S Chee, M. Mittmann, C. Wang, M Kobayashi, H. Horton, and E.L. Brown. Expression monitoring by hybridization to high-density oligonucleotide arrays. *Nature Biotechnology*, 14:1675–1680, 1996.

[44] K.H. Makambi. Weighted inverse chi-square method for correlated significance tests. *Journal of Applied Statistics*, 30(2):225–234, 2003.

[45] H.W. Mewes, K. Albermann, K. Heumann, S. Liebl, and F. Pfeiffer. Mips: A database for protein sequences, homology data and yeast genome information. *Nucleic Acid Research*, 25:28–30, 1997. http://mips.gsf.de/.

[46] L. Picard M.K. Kerr, E.H. Leiter and G.A. Churchill. Analysis of a designed mircoarray experiment. *Proceedings of the IEEE-Eurasip Nonlinear Signal and Image Processing Workshop, June 3-6 2001*, 2001.

[47] F. Mosteller and J.W. Tukey. *Data Analysis and Regression.* Addison-Wesley, Reading, MA, 1977.

[48] F. Naef and M.O. Nagnasco. Solving the riddle of bright mismatches: Labeling and effective binding in oligonucleotide arrays. *Physical Review E*, 68, 2003. 011906.

[49] K. Natarajan, M.R. Meyer, B.M. Jackson, D. Slade, C. Roberts, A.G. Hinnebusch, and M.J. Marton. Transcriptional profiling shows that gcn4p is a master regulator of gene expression during amino acid starvation in yeast. *Molecular and Cellular Biology*, 21:4347–4368, 2001.

[50] W. Pand, J. Lin, and C.T. Le. How many replicates of arrays are required to detect gene expression changes in microarray experiments? a mixture model approach. *Genome Biology*, 3(5):research0022.1–10, 2002.

[51] G. Parmigiani, E.S. Garrett, R. Irizarry, and S.L. Zeger. *The analysis of gene expression data: methods and software.* Springer, 2003. Chapter 5.

[52] P.B. Patnaik. The non-central χ^2- and f-distributions and their applications. *Biometrika*, 36:202–232, 1949.

[53] D. Rajagopalan. A comparison of statistical methods for analysis of high density oligonucleotide array data. *Bioinformatics*, 19(12):1469–1476, 2003.

[54] R. Ranallo, K. Struhl, and L. Stargell. A tata-binding protein mutant defective for tfiid complex formation in vivo. *Molecular and Cellular Biology*, 19:3951–57, 1997.

[55] A. Reiner, D. Yekutieli, and Y. Benjamini. Identifying differentially expressed genes using false discovery rate controlling procedures. *Bioinformatics*, 19(3):368–375, 2003.

[56] V.A. Rhodius and R.A. LaRossa. Uses and pitfalls of microarrays for studying transcriptional regulation. *Current Opinion in Microbiology*, 6:114–119, 2003.

[57] D.M. Rocke and B. Durbin. A model for measurement error for gene expression arrays. *Journal of Computational Biology*, 8(6):557–569, 2001.

[58] B. Rosati, F. Grau, A. Kuehler, S. Rodriguez, and D. McKinnon. Comparison of different probe-level ansysis techniques for oligonucleotide microarrays. *BioTehcniques*, 36:316–322, February 2004.

[59] E.E. Schadt, C. Li, B. Ellis, and W.H. Wong. Feature extraction and normalization algorithms for high-density oligonucleotide gene expression array data. *Journal of Cellular Chemistry Supplement*, 37:120–125, 2001.

[60] M. Schena. *Microarray Biochip Technology.* Eaton Publishing, Natick, MA, 2000.

[61] M. Schena, D. Shalon, R.W. Davis, and P.O. Brown. Quantitative monitoring of gene expression patterns with a complementary dna microarray. *Science*, 270:467–470, 1995.

[62] J. Seo, M. Bakay, Y.-W. Chen, S. Hilmer, B. Shneiderman, and E.P. Hoffman. Optimizing signal/noise ratios in expression profiling: Project-specific algorithm selection and detection p value weighting in affymetrix microarrays. *Bioinformatics*, 2004. Advance Access published April 29 2004.

[63] C.M. Smith. Bioinformatics, genomics, and proteomics. *The Scientist*, 14, 2000.

[64] E.M. Southern. Detection of specific sequences among dna fragments separated by gel electrophoresis. *Journal of Molecular Biology*, 98:503–517, 1975.

[65] L. Stargell and K. Struhl. A new class of activation-defective tata-binding protein mutants: evidence for two steps of transcriptional activation in vivo. *Molecular and Cellular Biology*, 16:4456–64, 1996.

[66] R.O. Stuart, K.T. Bush, and S.K. Nigam. Changes in global gene expression patterns during development and maturation of the rat kidney. *Proceedings of the National Academy of Sciences*, 98(10):5649–5654, 2001.

[67] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub. Interpreting patterns of gene expression with self-organizing maps: Methods and application to heatopoietic differentiation. *Proceedings of the National Academy of Sciences*, 96:2907–2912, 1999.

[68] W.A. Thomasson. Unraveling the mystery of protein folding. Technical report, FASEB. www.faseb.org/opar/protfold/protein.html.

[69] V.G. Tusher, R. Tibshirani, and G. Chu. Significance analysis of microarrays applied to the ioniziing radiation response. *Proceedings of the National Academy of Sciences*, 98:5116–5121, 2001.

[70] J. van Helden, B. Andre, and J. Collado-Vides. Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *Journal of Molecular Biology*, 281:827–42, 1998.

[71] P. Vieu. A note on density mode estimation. *Statistics and Probability Letters*, 26:297–307, 1996.

[72] X.S. Wang, S. Ghosh, and S. Guo. Quantitative quality control in microarray image processing and data acquisition. *Nucleic Acids Research*, 29, 2001.

[73] P.H. Westfall and S.S. Young. *Resampling-based multiple testing: examples and methods for p-value adjustment.* Wiley, 1993.

[74] G.R. Willsky, L.H. Chi, and D.C. Crans. Identification of gene expression changes in skeletal muscle from diabetic rats corrected by oral administration of vanadyl sulfate. *Proceedings of International Symposium on Bio-Trace Elements*, pages 119–124, 2002.

[75] C. Workman, L.J. Jensen, H. Jarmer, R. Berka, L. Gautier, H.B. Nielsen, H.-H. Saxild, C. Nielsen, S. Brunak, and S. Knudsen. A new non-linear normalization method for reducing variability in dna microarray experiments. *Genome Biology*, 3(9):research0048.1–0048.16, 2002.

[76] Z. Wu, R.A. Irizarry, R. Gentleman, F.M. Murillo, and F. Spencer. A model based background adjustment for oligonucleotide expression arrays. *Journal of the American Statistical Association*, 99(468):909–917, 2004. http://www.biostat.jhsph.edu/ ririzarr/papers/gcpaper.pdf.

[77] M.C.K. Yang, J.J. Yang, R.A. McIndoe, and J.X. She. Microarray experimental design: power and sample size considerations. *Physiol Genomics*, 16:24–28, 2003.

[78] Y.H. Yang, M.J. Buckley, S. Dudoit, and T.P. Speed. Comparison of methods for image analysis on cdna microarray data. *Journal of Computational and Graphical Statistics*, 11:108–136, 2002.

[79] L. Zhang, M.F. Miles, and K.D. Aldape. A model of molecular interactions on short oligonucleotide microarrays. *Nature Biotechnology*, 21(7):818–821, 2003.

[80] A. Zien, T. Aigner, R. Zimmer, and T. Lengauer. Centralization: a new method for the normalization of gene expression data. *Bioinformatics*, 17(Suppl. 1):S323–S331, 2001.

[81] A. Zien, J. Fluck, R. Zimmer, and T. Lengauer. Microarrays: How many do you need. *Journal of Computational Biology*, 10(3-4):653–667, 2003.