Image Analysis and Signal Extraction from cDNA Microarrays

Tracy L. Bergemann

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

University of Washington

2004

Program Authorized to Offer Degree: Public Health and Community Medicine - Biostatistics

INFORMATION TO USERS

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleed-through, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

University of Washington

Graduate School

This is to certify that I have examined this copy of a doctoral dissertation by

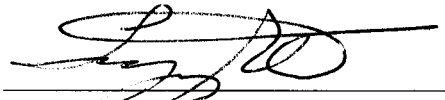
Tracy L. Bergemann

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

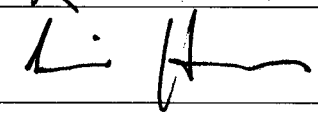
Chair of Supervisory Committee:

Lue Ping Zhao

Reading Committee:



Lue Ping Zhao



Ross Prentice

Li Hsu

Date: 10/28/04

In presenting this dissertation in partial fulfillment of the requirements for the doctoral degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of this dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for copying or reproduction of this dissertation may be referred to Bell and Howell Information and Learning, 300 North Zeeb Road, Ann Arbor, MI 48106-1346, to whom the author has granted "the right to reproduce and sell (a) copies of the manuscript in microform and/or (b) printed copies of the manuscript made from microform."

Signature Tracy L Briggeman

Date 10/26/04

University of Washington

Abstract

Image Analysis and Signal Extraction from cDNA Microarrays

by Tracy L. Bergemann

Chair of Supervisory Committee:

Professor Lue Ping Zhao
Biostatistics

The emergence of microarray technology invariably leads to a discussion about data reliability amongst researchers. Many factors impact the accuracy of gene expression data gleaned from microarray experiments. These factors range from noise inherent in the technology platform to variation arising out of experimental design. High level sources of variation encompass deviations that derive from the experimental design while low level sources of variation encompass the noise due to technological errors and biases in the lab. Although so-called high level sources of variation are dominant in microarray data most of the time, these sources can be over-shadowed by data errors at the technological level. That is, although differences in tissue sampled will likely cause most variation, images with artifact noise or dust covering information will taint small, but potentially crucial, sections of the dataset.

Because variation due to experimental design is well-covered territory in statistical research, the focus of this dissertation is at the low level of variation. The means to correct for sources of technological variation is not obvious to genomics specialists and statisticians. The research presented here explains the causes for cDNA microarray data variability and methods to account for the low level variance at two points: (1) image analysis and (2) signal extraction. The image analysis takes a TIFF image and performs grid alignment,

spot detection, background estimation, flagging and outputs the information to a text file. The image analysis routine to be outlined herein is automated, reproducible, and robust.

Signal extraction involves the modeling of spot pixel data to describe the overall spot intensity level and a measure of spot reliability while incoporating both red and green channels from the experiment. The spot quality measure will be spot-specific and continuous such that each data point in a set of experiments has an assigned data reliability weight. This quality measure can then be used to downweight low quality data in a regression-type analysis. In this way, spots that are tainted with artifact noise, and therefore have inaccurate expression levels, do not mar downstream analysis. A spot quality measure is also better than a flag, as summarily removing flagged data results in missing data problems. But using a spot quality weight does not result in missing data and may improve efficiency in test statistics.

A wide variety of methods to describe spot level quality estimates were investigated. The examination included several ways to incorporate spatial structure between pixel pairs. Semi-parametric methods to describe the variance of spots were not estimable for this data structure in the absence of spot replication. If updates to microarray technology protocols include spot replication, then semi-parametric measures can be revisited. Smoothers to describe correlation required a priori knowledge of the correlation size in order to adjust bandwidths resulting in a circularity problem. Ultimately, a fully parametric and a fully non-parametric estimate to describe quality are introduced and shown to be feasible for a data reliability model.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

Many thanks go out to Lue Ping Zhao, Steve Self, and Li Hsu for their guidance of my dissertation work and helpful suggestions. Additional thanks are due to Thomas Lumley, W. Whipple Neely, and Richard Laws for their contributions. The Institute for Pure and Applied Math at UCLA provided funding for part of this work and a fantastic environment to learn a great deal about genomics. Najma Khalid and Trina Brown deserve enormous kudos for their emotional support. Without them, I might have cracked. My gratitude to my parents, Allen and Allison Bergemann, for keeping me on track and listening to my weekly testimony of whining and doubt.

DEDICATION

This work is dedicated to my brother, Jason Bergemann, because we will always be there for each other and I could not have done this without him.

Chapter 1

INTRODUCTION TO MICROARRAY TECHNOLOGY

1.1 Introduction

The completion of the human genome map in recent years has provided a plethora of information about the chemical makeup of the human genetic code [49, 32]. Vast advances in biotechnology have increased the throughput capacity to access such information. Hence biomedical science has gone from an information poor state to an information rich state. But despite the rapid availability of coding sequences, there is much to learn about what parts of the sequence do, how they interact, when they are transcribed into mRNA, and when mRNA is translated into proteins. To gain insight into functional knowledge about the human genetic code, one approach is to examine expressed transcripts on a genome-wide level, leading to the development of the field of functional genomics [27]. Recently developed tools that facilitate such research are the various microarray technologies.

The human genome project is a massive international effort that began nearly two decades ago. Original funding and support came from the United States Department of Energy (DOE). Fractionation of ideals resulted in the development of two separate projects, one public and one private. Craig Ventor and his company, Celera, fronted the private human genome sequencing project. Eric Lander, now of the Whitehead Institute, fronted the public effort funded by the the DOE and the National Institute of Health (NIH). Improvements in sequencing technologies resulted in the completion of the sequencing effort ahead of schedule. Researchers assembled BAC (bacterial artificial chromosome) libraries of overlapping contigs to use as the base fragments for the sequencing effort. The fragments

were sequenced using in vitro DNA synthesis in the presence of chain terminating nucleotide triphosphates (ddNTPs). That is, DNA polymerase is used to construct complements to the base fragments. These complements are capped with a ddNTP. After all fragment complements are built and capped, they are separated by molecular weight. Each of the four ddNTP have fluorescent markers to differentiate between A, C, G, and T oligonucleotides. Algorithms can then be used to reconstruct sequences using the four fluorescent markers and the separations by molecular weight [39, 36].

While the above methods have provided sequencing information for entire genomes, further technologies have been developed to enhance this information. SAGE and microarray technologies provide information about transcription rates from the DNA sequence. Using clones of known mRNA transcripts and EST (Expressed Sequence Tag) libraries, rates of transcription from the genome can be assessed by measuring hybridization to these clones.

Burgeoning proteomic technologies provide information about translational rates or levels of protein expression. The oldest of the proteomic technologies is 2-D gel electrophoresis. Proteins expressed in a given sample are separated on gels by pH gradient and molecular weight. Proteins can be identified from these gels using mass spectrometry [36]. Newer methods for protein separation include matrix-assisted laser desorption/ionization time of flight (MALDI-TOF). This method separates molecules by their charge-to-mass ratio and lends itself better to automated high-throughput systems. The company Ciphergen briefly popularized the use of SELDI-TOF technology, or so-called protein chips. This technology, while providing an imprint of the peptide spectra within a cell, does not allow for identification of the actual proteins expressed. A medium throughput technology is tandem mass spectrometry (MSMS), that allows for the measurement of protein expression levels and identification of those proteins. All of the above mentioned proteomic tools, and others unmentioned, struggle with irregular and inconsistent signal generation that requires significant and extensive data pre-processing before data analysis can ensue.

Microarrays, the focus of this dissertation, are usually used to measure the level of

transcription for a predetermined set of genes or ESTs in biological samples. Currently, the eukaryote genomes for which there is at least a draft of complete sequence information include S. cerevisiae, C. elegans, Drosophila melanogaster, Arabidopsis thaliana, Homo sapien, and Mus musculus [19, 17, 1, 28, 32, 14]. Hence microarrays supporting complete clone sets from any of these species can measure transcription rates at the genomic level for a given time and space. Knowledge of the transcription levels in an entire system can shed light on how the system responds to environmental stimulus or to gene variants within a species. Microarrays will provide clues about the function of many genes that have not yet been fully annotated and potentially how these genes interact with one another.

Knowledge of genomic transcription has great potential for use in drug development. That is, the array technologies could become useful for screening, toxicity determinations, and phase I and II clinical trials. Often, phase III clinical trials are stopped short due to a significant number of adverse events in the treatment population. Ending clinical trials for a potentially effective treatment that causes adverse effects in a small but significant population is a large financial setback for the pharmaceutical industry. If, however, pharmaceutical companies could predict those groups for which an adverse event is likely, those groups could be deemed ineligible. Hence the treatment of interest will be tested on the subgroup that is unlikely to suffer adverse events. And subsequent to FDA approval, the treatment will only be used in a clinical setting where the patient is unlikely to suffer ill effects from the drug. The pharmaceutical industry wagers that microarrays and other genomic biotechnologies can be used as a genetic tag to predict those small subgroups that will not respond to drug agents in a favorable way. This idea is currently being referred to as pharmacogenomics.

In addition, there are potentially powerful clinical applications for microarray research [11]. If transcription rates can be correlated with disease outcomes, medical professionals will have more accurate markers for these outcomes. Profiles of gene expression levels might even help to create further disease subsets with distinct endpoints. Treatment strategies can

be more narrowly targeted to those situations where it is necessary and suitable. A rationale for the possibility of these clinical applications is that expression profiles are a reasonable indicator of genome function and activity. And specific disease stages correspond with specific gene activity. Gene activity can include mutations leading to an overabundance or under-abundance of certain transcripts with the consequence of diseased status.

The focus of this work is array technologies. While knowledge of protein expression would be immediately more reliable in learning about cellular activity, technologies to describe protein expression remain limited. The next best step is to measure gene expression, as this provides clues about resulting protein expression and gene function. Further, gene expression at this time seems more promising than simple knowledge of genome sequence as no one yet has been able to "crack the code" and consistently predict gene function from sequence. Finally, as will be noted in Section 1.2, preliminary studies using microarray technologies suggest promise for further understanding of molecular biology and disease processes.

1.2 Technologies and Examples

1.2.1 Microarray Technologies

Microarray technologies fall largely into two camps: oligonucleotide arrays and two-color comparatively hybridized cDNA arrays. Commercially available oligonucleotide arrays such as GeneChip®, produced by Affymetrix, provide information about the levels of mRNA expression for several species. Expression is measured by synthesizing anywhere from 13 to 20 different probe sets of short 25mer oligos that represent different regions in a particular gene. The probe set sequences are available online to registered Affymetrix users. Affymetrix claims that the sets are representative and selected to maximize binding affinity to target mRNA sequence. The probe sets are built onto the GeneChip® surface using photolithography. Using a mask, each probe sequence is built up one oligonucleotide at a time. The technology is such that occasionally oligos will fall off and the built sequence is

incomplete. It is not possible, however, to determine which sequences are incomplete and how often this occurs on a particular chip.

For each probe set on an Affymetrix chip, there is a "perfect match" row and a "mismatch" row. The perfect match row contains sequences representing regions from the gene of interest. The mismatch row contains the same sequences from the same gene, but the 13th position of each 25mer probe is switched to its complement. The mismatch serves as a control, accounting for non-specific hybridization and other experimental noise. Thus, expression levels from the perfect match row are adjusted for using the levels from the mismatch row. Larger expression levels from the mismatch row resulting in negative gene expression estimates are a constant source of confusion and debate. Negative values are calculated for upwards of 25% of all genes tested and negative values are not necessarily consistent across arrays making the predication of their existence more difficult. Affymetrix provides an "idealized mismatch" in the latest version of their software that attempts to account for some of these irregularities.

Affymetrix provides software to process the images gleaned from their chips. Their image analysis software is aided by a highly regulated placement system that includes tags to help locate the regions of interest. Researchers are discouraged from performing their own image analysis of scanned Affymetrix chips and upgrades of the GeneChip® might make it even more difficult to perform an independent image analysis. That is, there has been some discussion of scattering the order of probe sets and keeping the order proprietary. Current prices for academic institutions and non-profit organizations, while about half of that for industry, still remain prohibitively expensive limiting its uses in large-scale experiments. In addition, analytic algorithms are proprietary, and hence my image analysis research focuses on custom-made cDNA microarrays, also known as spotted arrays.

Over the past few years, cDNA microarray technology has grown in its use and now serves as a valuable analytical tool for a number of functional genomics applications. A typical microarray assay involves the use of a library of unique sequence-specific cDNA

"clones" that are carefully selected to represent different regions (e.g., genes) of the genome of interest. Each clone is amplified into millions of copies via polymerase chain reaction (PCR) and deposited in an addressable, grid-like fashion onto a glass support using pen tip or ink jet spotting methodologies. The end result is an arrayed surface containing up to tens of thousands of features, 100-150 μm in diameter, each of which can contain millions of clones unique to a specific region of the genome. The spot features are grouped into blocks (or pen tip groups) on the glass support.

Spotted array technologies employ competitive hybridization techniques, requiring the use of two target tissues that are simultaneously queried in the assay. In each application, fluorescent-labeled targets are prepared using mRNA or genomic DNA and are introduced onto the arrayed surface under conditions that promote selective hybridization. The two tissues of interest, typically a reference and experimental state are differentially labeled with fluorescent dyes and co-hybridized to the array surface, where the reference target serves to normalize the experimental target signal at each feature on the array. Accordingly, the results for features are reported as a ratio of expressions levels. The subsequent scanning of a post-hybridized array produces an image (see Figure 1.1 for a sample image) with varying degrees of fluorescent intensity for each spotted feature, as well as the background fluorescence corresponding to the arrayed surface. As such, an appropriate analysis of an array image is instrumental to the biological understanding of the experiment under investigation. Spotted array experiments will be discussed thoroughly in Section 1.3.

1.2.2 Uses for Microarray Technology

Now that the technologies have been briefly described, the motivation for the intense interest in this technology should be further explained. Microarrays measure transcription. Transcription is the process by which a DNA sequence has its message carried outside the nucleus where it will then be translated into a protein. The transcript or mRNA is complementary to the original DNA sequence. The level of transcription or expression gives us

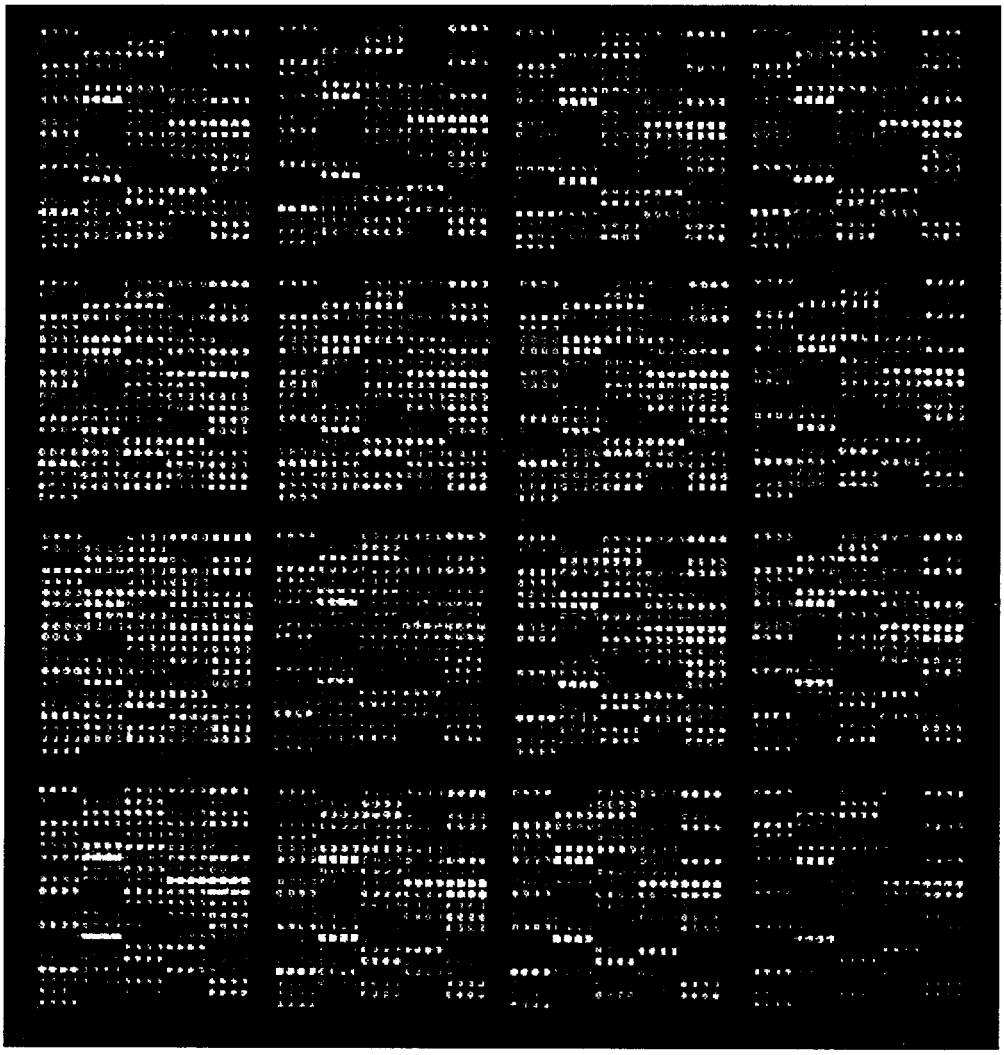


Figure 1.1: A quality image representing comparative hybridization of mRNA extracted from yeast cells.

an idea of how often a DNA sequence will be converted to a protein for a given tissue at a given point in time. These proteins will then perform all the functions necessary for life such as energy production and chromosome reproduction. Usually the level of expression is regulated by transcription factors that respond to the cell environment. Transcription varies dependent on cell cycle, conditions of resource deprivation or abundance, environmental extremes, etc. Hence, if we can predict expression, we come closer to understanding gene function at a genomic level.

The following sections give important examples from the early literature about the uses of the microarray technology. A veritable flood of papers have been published since applying microarray technology to nearly every nook and cranny of the biological sciences.

Feasibility of Technology

One of the earliest papers published using microarray data came from the Stanford lab [43]. Forty-five arabidopsis genes were tested for their expression in three array experiments. As one of the first papers about spotted arrays, its focus was on the technology and how the experiment was conducted. A HAT4 transgenic plant was compared to wild-type with the expected result being over-expression of HAT4 in the transgenic strain. No other genes were found to be differentially expressed in that experiment, but the tested genes comprise less than 0.3% of the arabidopsis genome. A second experiment compared gene expression in leaf tissue versus stem tissue and identified 26 genes with expression differing by more than a factor of five. Again, as one of the earliest papers on microarrays, the intent was not to test a biologically driven hypothesis, but rather to test the feasibility of the technology.

Glucose Pathways in Yeast

Later papers have of course been more targeted in their genomic studies. Microarrays have been used to examine transcription of the yeast genome during the shift from anaerobic to aerobic metabolism [16]. Yeast in a sugar rich medium will grow rapidly and produce

ethanol (anaerobic growth). When the glucose source is exhausted, yeast use the produced ethanol source instead (aerobic growth). This is called the diauxic shift. Variations in gene expression during the diauxic shift were monitored using microarrays. Because many of the metabolic pathways in yeast are already understood, expression patterns of genes known to be involved in these pathways could be mapped onto them. In this way, the experiment tested whether the measured gene expressed as expected. Genes of unknown functions were clustered with genes that had similar expression patterns. Assuming that genes with similar expression carry out similar function, genes with unknown function could then be assigned their likely role in a metabolic pathway.

Synchronized Cell Cycles

The Stanford array group has also examined other facets concerning function of the yeast genome [47]. Experiments were performed on various yeast strains and expression was monitored for various time points in the cell cycle. Yeast cells were synchronized, i.e., the cell cycle was arrested and then restarted so that all cells enter each phase of the cycle at approximately the same time. Of interest were those genes with expression patterns regulated by the cell cycle. Cell division is a self-regulating program meaning that many genes controlling the cell cycle are also controlled by it. Roughly 800 ORFs were identified with transcription dependent on cell cycle. Researchers examined genes of known function for peak expression levels at various points in the cycle. Cluster analysis was performed in an attempt to group genes into functional categories such as DNA replication, repair or assembly, budding, mitosis, mating, cell cycle control, etc. In this way genes of unknown function are inferred into a category by similarities in expression. The hope is that cluster analysis will "provide a foundation for understanding the transcriptional mechanisms of cell cycle regulation". Genes within the same clusters were examined for motifs in the promoter region. Often, there was strong indication for common motifs within each cluster and common motifs for genes peaking at the same phase in the cell cycle. Potentially, this

means that co-regulated genes share the same regulatory transcriptional elements. Separate experiments were conducted to examine the effects of inducing the G1 cyclin CLN3 and the mitotic cyclin CLB2. Reported findings indicate that about half of all cell cycle regulated genes can be controlled by CLN3 or CLB2 cyclins.

Leukemia Classification

One of many goals of cancer research is to classify disease into sub-categories with prognostic significance. This has long been the work of pathologists examining stained slides for patterns in cellular structures. While the field of pathobiology has contributed much to the classification of cancer sub-types, many important differences in disease pathology may only be seen at the molecular level. Instead of examining tumor cells for patterns, we can now look at expression profiles for patterns. One of the first papers with this goal in mind used expression profiles to classify acute myeloid leukemia (AML) and acute lymphoblastic leukemia (ALL) [20]. Correct classification of these two acute leukemias is important as effective treatment strategies markedly differ. Experiments were conducted using Affymetrix arrays to measure expression in bone marrow samples of leukemia patients known to have either AML or ALL. Fifty genes were selected to construct a classification rule based on expression levels in 38 patients. The classification rule correctly identified 36 of the 38 samples. This rule was then applied to a new set of data from 34 patients, classifying 29 of the 34 appropriately. This initial paper indicates the potential that microarrays have to classify tumors into categories with prognostic significance.

DNA Copy Number Changes

Loss of heterozygosity (LOH) studies in relation to cancer research have been going on for several years. Cancer is often associated with gross chromosomal abnormalities and changes in DNA copy number production within the nucleus. The significance of these changes and the ramifications thereof remain active questions of interest. The study of changes in copy

number at particular loci can often lead to the discovery of oncogenes. Standard methods to detect genome-wide DNA copy number changes include comparative genomic hybridization (CGH) and fluorescence *in situ* hybridization (FISH). Because CGH has a limited mapping resolution and FISH is labor intensive, microarrays are being considered as an alternative [42]. Clone sets consist of cDNA from radiation-hybrid mapped genes, a higher resolution map than for CGH. Cell lines from cancer types of interest are co-hybridized with normal tissue. The resulting ratios from the microarray experiments provide evidence for change in DNA copy number. If the ratio is significantly less than one, it corresponds to DNA loss and if the ratio is significantly greater than one, it corresponds to DNA gain. The ratios can be compared to the radiation-hybrid map to determine chromosomal regions with deviation in DNA production. As microarray technologies improve, they are likely to become another standard method for examination of copy number changes.

Identical by Descent

Our last example of microarray use from published research performs identical-by-decent (IBD) mapping employing genomic mismatch scanning (GMS) [10]. The goal here is to locate regions less than two megabases long containing disease genes of interest. This requires less effort than linkage studies genotyping several large pedigree families on a sparse marker set. Instead, unrelated diseased individuals are chosen from the same founder population. These individuals are compared to determine regions conserved in their genomic sequence. With a large enough number of comparisons, the conserved sequence will shrink to a manageable and searchable region likely to include a gene of interest. Cheung et al chose Ashkenazi Jews as their founder population and hyperinsulinism as the disease of interest. The causal gene for this disease is already known to lie on chromosome 11p15.1. Hence, it was of interest to recover this information using new techniques. Unrelated pairs of affected patients were selected and screened using GMS to tag sequences shared between the two people. These sequences were hybridized to an array containing YAC clones from

chromosome 11. Conserved regions were identified by observing those clones that hybridized to tagged sequences from the pairs. Eight independent and unrelated pairs were used to successfully recover the known disease region. This success indicates that further studies using arrayed clones from the entire human genome may be able to identify disease genes in a founder population.

1.3 Sources of variation in Microarray Experiments

1.3.1 A Typical Experiment

The following section includes information about microarray experiments drawn from a variety of sources. Much of the information came from personal communications with Jeff Delrow of the Fred Hutchinson Cancer Research Center, Berry Merriman from Stan Nelson's lab at UCLA, and Jobst Landgrebe of the Max Planck Institute for Psychiatry. Official and widely used resources for building array facilities include the M-Guide available at Pat Brown's website (http://cmgm.stanford.edu/pbrown/mguide/index.html) and the Cold Spring Harbor Protocols available at http://www.microarrays.org/protocols.html.

To illustrate microarray experimental procedure, an example framework will be provided here inspired by ongoing genomics research of the molecular biology of Wilms tumor. In cancer research, treatment strategies are often based on the assessment of risk factors for favorable treatment response. For the treatment of Wilms tumor, a rare malignant renal tumor afflicting children, this targeting strategy has been particularly successful. Five year survival for Wilms tumor patients is now close to 90%. Much of this success can be attributed to a well-defined staging system and the identification of the anaplastic histology [4]. Anaplastic Wilms tumors account for half of all Wilms deaths. Nevertheless, other prognostic histologic factors of importance have not been identified and there are still many Wilms tumor deaths that are unexplained. For example, it is known that only about 30% of stage III and IV Wilms tumors benefit from adding doxorubicin to the chemotherapy regimen. As doxorubicin is associated with an increased risk in congestive heart failure,

detecting those patients that do not require doxorubicin for recovery would be useful [22].

Current thinking suggests that those factors not identifiable by pathobiologists at the cell level will be identifiable at the molecular level by examining transcription patterns. Experiments could be conducted using microarrays to help identify molecular biomarkers of prognostic significance. For example, mRNA could be extracted from two stage IV metastatic tumors of favorable histology, one for which the patient responds well to doxorubicin treatment and survives cancer for at least five years and the other for which the patient does not respond to doxorubicin treatment and suffers a negative outcome. If this experiment was repeated several times for several patients, the average differences in genetic expression between the two groups could be estimated. Those genes that consistently express at different levels for the two comparison groups would be identified as potential biomarkers to predict poor prognosis. Patients exhibiting abnormal expression patterns for an identified biomarker could then be targeted with more aggressive therapy such as the addition of doxorubicin. Therapy can be alleviated for those patients with good prognosis. In this way, prediction of poor prognosis for Wilms tumor could be further refined and subsequently, treatment strategies can also be refined, leading to a higher quality of life for some patients.

Initial stages of microarray experiments involve selecting the genes or ESTs for which the level of mRNA expression is of interest. For each sequence selected for testing in experiments, sequence-validated cDNA clones are obtained. These clones are amplified via PCR to generate sufficient content for comparative hybridization. Individual PCR products are verified as unique via gel electrophoresis and then purified. Next, the purified clone products are transferred to wells containing buffer solution on microtitre plates from which samples will be extracted by robotic pin-tips.

After selecting and preparing clones for testing, they are placed onto glass slides via a printing process. Initially, poly-lysine is used to coat glass slides because it facilitates adhesion of the PCR products onto the glass. The slides are affixed securely (i.e., taped

down) to the arraying table in order to avoid shifting in the ensuing printing process. The microtitre plates contain either 196 or 384 products and hence, several plates are necessary in order to print the thousands of different kinds of clones. Pins, which look and act much like a fountain pen, will be used to dip into the plate wells, collect roughly 0.5 μl of product, and place a few nl of product onto each microarray slide. Pins are blotted to remove excess product before being arrayed. A robot is used to ensure that clone placement for each sequence is equidistant on the slide. The robots typically hold 16 pins at a time. Each pin is responsible for its own quadrant, or block, on the array. The robot is also used to wash and dry the pins before they are dipped into the next wells on the microtitre plate and print the next product.

In addition to preparing clones for testing, target tissues of interest need to be selected to compare their transcription levels. Once tissues have been chosen, mRNA is extracted by constructing a poly-T sequence and using this to attract the poly-A tail from the messages. Alternatively, mRNA could be isolated from total RNA extracted from the tissues of interest. The reverse transcriptase enzyme is used to convert the mRNA to cDNA for each tissue type. These cDNA strands are labeled either with cy3 (green) or cy5 (red) dyes. That is, strands from one tissue type should be labeled with the cy3 fluor while the other tissue type is labeled with the cy5 fluor. Several labeling protocols are used in microarray experiments. The mostly widely used protocol involves direct fluorescent dye incorporation using cy3- or cy5-dUTP in the target mRNA. Let it be noted that it is possible for more than one fluor to bind to one sequence. This could result in a bias unless the occurrence of multiple labeling happens with similar rates for both dye types. Other, newer, labeling protocols such as the amino-allyl label involves the construction of a sequence tag, aa-dUTP, at the end of target mRNA sequences followed by cy3/cy5 coupling with the tag.

Before the hybridization step, many labs heat their microarray slides to denature the cDNA clones. This allows for hybridization reactions between the double stranded clones and the single stranded targets. The author is not certain that all labs perform a denaturing

step before hybridization. If no denaturing step occurs, it is unclear how hybridization takes place. Drops of saline solution containing the labeled target cDNA strands are placed onto a printed microarray slide and covered with a glass slip. The microarray slides are placed in hybridization chambers and kept in a $63°C$ water bath. The hybridization chambers help to keep environmental conditions constant. The slides are left for roughly six hours in order to allow for all possible hybridization reactions. Once all possible reactions are thought to be complete, the arrays are removed from the hybridization chambers and undergo several washings. Washing the arrays results in the removal of extraneous sequences that did not hybridize properly. After the arrays have been washed in two different solutions, they are dried in a room temperature table top at 600 rpm for five minutes.

The final step in the experimentation process involves the conversion of the hybridized microarray to a digital image. The microarrays are scanned for two wavelengths to fluoresce the cy3 and cy5 dyes. A laser shoots through an objective lens with an excitation wavelength of 532nm for the cy3 dye and 635 nm for the cy5 dye. The emissions from the laser excitation are recorded through an objective confocal lens using either a CCD camera or a photomultiplier tub (PMT). This quantification is stored as two TIFF images, one for each dye type, consisting of 2-D digitized matrices of 16-bit intensities. Currently, scanners deviate in the process by which they deal with multiple channels. That is, some scanners will process the microarray first for one channel and then for the other. Other scanners process microarrays simultaneously for both channels. The most popular scanner, produced by Axon (Axon Instruments, Inc. Foster City, CA), has this capability. Scanning arrays simultaneously result in two images that overlap more precisely.

1.3.2 High level analysis

Here we discriminate between "high" and "low" level analysis of variation. Low level analysis pertains to image analysis, measurement error, and signal extraction. In short, low level analysis involves the generation of a clean data set. High level analysis pertains to the

Table 1.1: Steps in a cDNA Microarray Experiment

Step	Process	Details
1	Clone Selection	Sequence validated clones from genome of interest
2	Amplification	PCR to produce millions of clone copies
3	Clone Placement	Arrangement of clones in microtitre plates
4	Printing	Robotic arm spots clones with a pin-tip configuration
5	Tissue Choice	Selection of cells of interest
6	RNA extraction	mRNA or total RNA is taken from cells
7	Labeling	AA-dUTP or amino-allyl attachment of cy3/cy5 dyes
8	Hybridization	Comparative binding of matching extract/clone sequences
9	Washing	Removes non-hybridized sequence
10	Drying	In preparation for scan
11	Scanning	Excitation of dyes and conversion to digital images

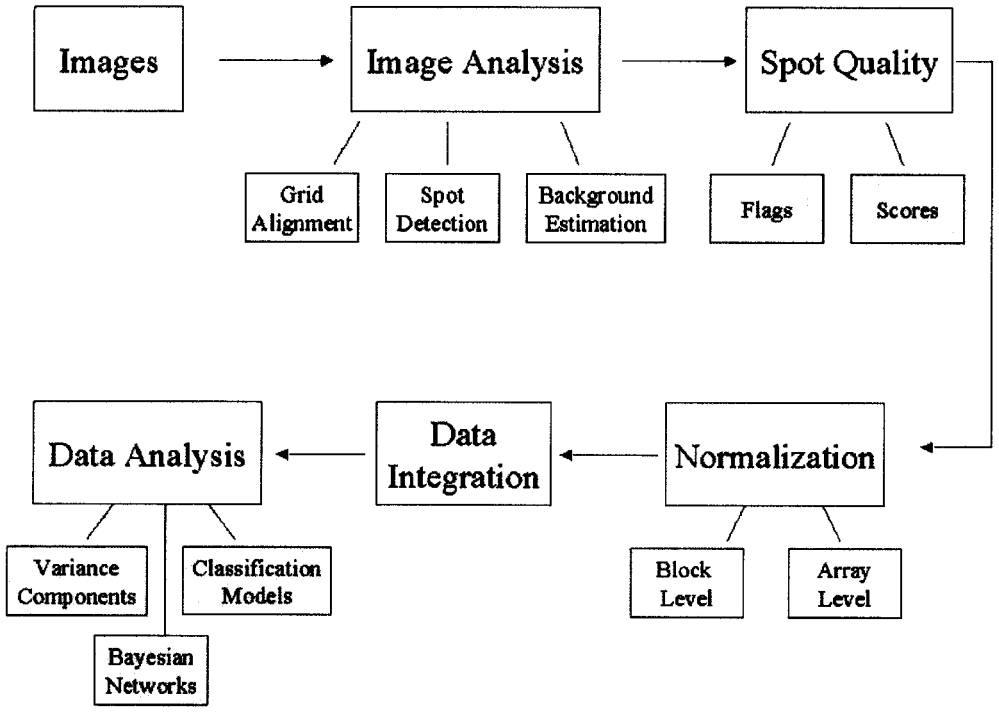


Figure 1.2: The work flow in cDNA microarray data analysis.

analysis performed on the data set, such as the search for differentially expressed genes or classification of tumor tissues by expression profiles. Naturally, the high level analysis depends on the quality of the low level analysis. The focus of this dissertation will be the low level analysis of microarray variations. But first we will briefly discuss the other.

Classic experimental design methods developed by R.A. Fisher are as relevant to microarray experiments as they were in Fisher's agricultural studies. These methods acknowledge that experimental factors contribute to bias when accounting for variation. For example, one microarray image might have been generated under slightly different conditions than another. Some labs have already examined these experimental variations using classical methods [30, 31]. A linear model or ANOVA is constructed as follows. Let A_i represent a microarray experiment, G_j represent each gene measured per experiment, T_k represent the tissue type or treatment effect (usually $k = 2$), and D_d represent the dye label where $d \in \{0 = \text{red}, 1 = \text{green}\}$. A linear model can then be constructed to explain variances in gene expression examining main effects, two-way interactions, three-way interactions, and four-way interactions in the saturated model. For example,

$$\begin{aligned} Y_{ijkd} &= \mu + A_i + G_j + T_k + D_d \\ &\quad + (AG)_{ij} + (AT)_{ik} + (AD)_{id} + (GT)_{jk} + (GD)_{jd} + (TD)_{kd} + (AGD)_{ijd} \end{aligned}$$

where Y_{ijkd} is the spot intensity value of gene j for treatment k on array i using dye d. In this setting, our variable of interest is $(GT)_{jk}$, representing change in a gene's expression for the treatment of interest. Even in this model using only one three-way interaction, it is likely that insufficient degrees of freedom and lack of balance will result in inestimable quantities. Nevertheless, it is likely that such three-way interactions will be significant in microarray experiments and hence can not be ignored. The current practice, however, is to model only some of the two-way interactions and none of the higher order interactions. This is one problematic assumption, to say nothing of the other assumptions about normality, linearity, and independence.

Alternative models exist to measure sources of variation that assume less. Models using estimating equations relax the independence assumption [48, 56]. Models that use an alternate outcome variable can reduce the number of factors and interactions included:

$$\log \frac{R_{ijk}}{G_{ijk}} = \mu + A_i + G_j + T_k + (AG)_{ij} + (AT)_{ik} + (GT)_{jk} + (AGT)_{ijk}$$

here R_{ijk} is the spot intensity in the red channel for array i, gene j, and treatment k and G_{ijk} is the complement in the green channel. Using a log-ratio as the response introduces new assumptions and modifies the model's interpretation, but also reduces the factors involved and allows for simpler estimation.

It is worth noting in this section that the reference sample in cDNA microarray experiments plays an important role in experimental design. The simplest and most common type of reference sample is from a supposedly "normal" batch of tissue cells. This reference is compared to a diseased batch, for example, cancerous cells. But other researchers are using the same reference material for all of their array studies, that is, a global reference. Two common variants are to either pool mRNA from tissues taken from many different organs within the organism of interest, or to use mRNA from a altogether different species than the organism of interest.

Many other published methods exist for the analysis of microarray data, of a breadth and variety too large to cover here. An exhaustive review of the methods would create material for an entire book and will not be attempted here. Regardless of model specification and experimental design, the estimation procedures will be affected by the reliability of the data extracted in the first place.

1.3.3 Low level analysis

This section will give attention to the sources of error within a single microarray experiment. Some of these sources have been rather well discussed such as the differences in labeling efficiencies for the two dyes used, that calls for a "normalization" procedure. Or the presence

of block effects, sometimes known as pin tip effects, will result in overall differences in gene expression from one block to the next within a microarray. Block effects, being completely confounded with pin tip, might encompass variations due to sources other than pin tips. Hence, it is best to call these effects block effects. There is also the potential for block:array interactions where effects are different between blocks and also between blocks on different arrays. Block variations are important because they potentially provide information not only about spatial variation within an array but also about variations due to each pin tip in the spotting process.

Other sources of variability have been discussed little to not at all. Spatial variations that are not accounted for by block groups are also important to accommodate. Particularly if these variations are irregular or non-linear, a simple blocking factor will not resolve these deviations. Methods from spatial statistics have as of yet not been applied to microarray images. Another variation that has been studied little to date is the within spot variance. Each spot on an array image is comprised of hundreds of pixels. These pixels often have intensity values vastly different from one another within the same spot. Within spot variation may be important because it provides an idea of how variable the gene expression measurement is for a given experiment. Further, the large number of pixels present within each spot will allow for decent estimation of within spot variation. These estimates can be used downstream in high level analysis thereby accounting for the reliability of each spot estimate.

When analyzing microarray images, several statistical issues must be breached. While most first order problems have already been categorized fairly well, such as how to quantify the intensity of each spot on the array, some first order problems remain. One of the problems that remains unresolved at this time is how to quantify background noise on the array, i.e., how to measure the "background". Also, there is much debate as to how to measure the relative intensity of each spot for the two tissue samples involved. A ratio seems to be the most obvious choice to quantify the relative intensities, but the ratios are accompanied

by several problems. In particular, ratios become unreliable, approaching infinity or zero, when the level of transcription is low in one of channel on the array. Second order problems concern the estimation of variability, which have not received much attention at this time. However, assessing such variations is important, because such values indicate the quality of extracted signals and such information should be incorporated in the downstream analysis along with extracted signals. To achieve this objective, one needs to study random variations of pixel-specific intensity values and to take into account spatial dependencies in variance calculations. Hence, work must be done to develop stable descriptions of relative intensity levels, to agree upon the measure of background noise, and to quantify variability in intensity levels.

While the above mentioned targets for variation in expression response can well be accounted for with statistical methods, other sources of variation can not be so easily be measured as they arise from biases in the experimental process. We will differentiate these biases by calling them technological variations as opposed to experimental variations. As shown in Table 1, microarray experiments involve many steps. At each step there is the potential for systematic bias that must either be adjusted for in the statistical model or corrected with improved technology. The remainder of this chapter is devoted to delineating the known sources of error for each step in a microarray experiment.

The first generation of spotted array technology often used clones that had not been validated for sequence content. This was an egregious error indeed because it meant that genes the researchers intended to measure were not necessarily the genes actually being measured for their expression! A study of clone purity conducted three years ago indicated that as little as 62% of stocks represented a pure sample of the correct clone [24]. Stocks containing more than one type of cDNA likely result from cross-contamination or handling errors. But as gene sequencing technology has improved so has the researchers' ability to obtain representative clones. Further, it is becoming common practice is to validate gene clones for sequence content to avoid errors. Amplification of the clones is also relatively

painless as PCR is a well-developed technique used millions of times per day.

The third step involves clone placement. Amplified clones precipitate overnight, are resuspended in buffer solution, and then transferred onto microtitre plates. Some buffer solutions serve better than others to maintain the integrity of the bonds in DNA sequences. The most popular buffer is 3X SSC, but some labs are now using organic salts as buffers and claim to get different hybridization results. The point here is simply that the buffer solution used might influence the endpoint and as such, is a source for technological variation. Another concern with clone placement is tracking which clones are kept in which plate well. This is a database and documentation matter. Considering that experiments often use thousands of different clone sequences arrayed in 192 or 384 well plates, it is not trivial to track a clone from the freezer through the PCR amplification to a well on a microtitre plate to its final destination on a microarray slide. Plates can be mixed up, clones could be placed in the wrong row or column, etc. Any mistake in clone placement would result in faulty data, and once the clones have been placed onto an array, there is no easy way to tell if the clone is representative.

Clones are printed onto microarray slides using a robotic arm. Feasibly, about one hundred slides can be printed in a run, depending on the size of the printing table. The robotic arm carries pin tips of two possible varieties. One variety, as mentioned before, functions essentially as a fountain pen whose tips must first be blotted to remove excess solution before printing. The other variety is a solid pin tip that does not require blotting in order to print equal amounts of solution. Naturally, as the robot arm moves from one array to the next printing its set of clones, the droplet placed differs in size from one array to the next. The deviation in droplet size will usually be in nanometers, but this will result in a notably different amount of cDNA placed onto each spot per array. Because two tissues will be comparatively hybridized to each spot, the amount of cDNA should not matter greatly and will be adjusted for with a ratio. If the amount of printed cDNA is small, hybridizations will be difficult to detect. And it is not completely clear that a ratio accounts for differences

in spot size. The differences in spot size also make it difficult to perform image analysis. Further, slight shifts in movement during the printing process will result in spots grids that are slightly askew. This again will affect image analysis procedures since most packages rely on grid alignment to help find spot regions. If the grid alignment is askew, spot detection will be more difficult.

The next variable discussed is both an experimental and technological source of variation. Selecting the tissues to be used for comparative hybridization is arguably the most important decision in the experimental design process. The tissues selected must be the most appropriate to answer the scientific question of interest. The subject of tissue selection is an article in and of itself, and has been, particularly in cancer research and expression profiling of cell-cycles [13, 18]. In the Wilms tumor example given above, researchers wish to predict outcome from gene expression profiles within the same stage and histology. This requires the selection of two groups of tumors from the patient population that are as similar as possible but for which the groups of patients experienced opposite outcomes. Pathologists can testify that this is not a trivial task, however. Most tumors are heterogenous in nature, with several histological patterns existent within the same tumor biopsy. Thus, to compare tumor tissues from two different patients of the same histology, cancerous cells need to be extracted that are of the same histology. Given, the mixed nature of most of the patient population's tumors, this is quite challenging.

Once tissues have been selected for experimentation, mRNA is extracted. mRNA degrades outside the nucleus and is not necessarily translated into protein. Tissues that have been frozen, common in prospective studies, are subject to mRNA degradation of unknown quantity. Hence mRNA extracted from selected tissues is not necessarily equal to the amount of message that was transcribed within the nucleus. If an inadequate amount of mRNA is extracted, the sequences are amplified using PCR. In a perfect world, PCR is a copy machine with production rate $f(r) = s * 2^r$ for every sequence in the sample where r is the reaction number and s is the number of starter sequences. There is a bias, however,

toward sequences most prevalent in the sample. If for Gene One you have 10 starter sequences and for Gene Two you have 100 starter sequences and 10 reactions are performed, you should end up with 10,240 copies of Gene One and 102,400 copies of Gene Two. In a hybridization experiment, Gene Two will be much more easy to detect than Gene One because there are 92,160 more copies. Some labs are now using linear PCR to account for the bias toward more abundant sequences, but the details have not been ironed out.

Labeling involves coupling of dyes to extracted mRNA sequences via a chemical bond. Traditionally the dyes bond to one base type, such as cytosine and hence there can be more than one dye attached to a sequence. The binding affinities of cy3 and cy5 are known to differ. The adjustment for differences in labeling efficiencies is known, perhaps inappropriately, as normalization. There does not seem to be a hard and fast rule about the differences in binding affinity for the two dyes. This seems to vary from lab to lab and experiment to experiment. Thus, normalization also varies from lab to lab and often from experiment to experiment. New protocols are being developed to adjust for differences in labeling efficiencies. The amino-allyl protocol mentioned earlier is one such strategy. Because the labeling in this protocol is not sequence dependent, each sequence of interest receives no more than one label. This does not eliminate the potential bias due to greater binding affinity of one dye over the other and hence this must still be adjusted. Dye contamination for the cy5 fluor has been observed by some researchers using hyperspectral imaging. This contamination is specific to the green channel and appears only within the spot region, indicating artifact noise in the binding of the green fluor to sequences [37].

The hybridization process is fairly understood in the "wet lab" environment when the number of different types of sequences tested per experiment are small. The microarray experiment calls for hybridization in a dry environment. Very little is understood about this state. Equations for molecular kinetics or time to hybridization are not known. When microarrays are not subjected to a denaturing step, it is also not known how double stranded cDNA clones manage to couple with the single stranded mRNA extract. Hybridization

occurs at $63°C$, well below the temperature required for denaturing. Presumably, the double helix structure is not rigid and therefore provides an opportunity for the mRNA extract to bind to its complement clone sequence. So not only are the two tissues vying for clone sequences to bind to, they also have to compete with the complement sequence that is attached to the probe of interest.

The dirty secret of spotted array experiments is that hybridization is not fully specific. Samples from the mRNA extract can potentially bind to non-complementary clones. Non-specific binding occurs in microarray experiments because the optimal hybridization environment is sequence dependent. Trying to create an environment conducive to the hybridization of 10,000 different sequences is equivalent to the non-linear optimization of a 10,000 dimensional problem. Non-specific binding is currently not accounted for in microarray experiments as there is no readily available method to measure this phenomenon. The lack of specificity is also known to vary with sequence length. Using 25mer probes as Affymetrix does can result in highly sensitive hybridizations that are not very specific. On the other hand, if the sequence is very long as in the spotted arrays using cDNA strands from 500 to 1000 base pairs long, cross-hybridization becomes a problem. Personal testimony by array experimenters is that cross hybridization is common for two sequences over 75% identical [47]. Southern et al further discuss the relationship between sequence length and cross hybridization [46]. The moral of the story here is that because so little is known about the hybridization process during a microarray experiment, there are potentially many unmeasured sources of variation at this step.

Arrays are washed after hybridization to remove mRNA extract that did not bind to sequence clones on the microarray. It is possible that some hybridized strains will be removed during the washing process. Particularly, mRNA extract that does not bind well with its complement might be removed in this process. The array bath can potentially leave water marks on the slide that add artifact noise when performing an image analysis.

Drying the microarrays involves spinning them on a table for several minutes. The

drying process often results in the famous comet tails seen on many early array images. That is, while the arrays are rotating, hybridized gene product can leak out onto the array surface forming streaks. These comet tails make it difficult to segment the array image and determine what constitutes signal versus background.

Scanning involves the excitation of fluorescent dyes using two different wavelengths. Note that although each dye responds most easily to a given wavelength (532nm for green, 635nm for red), the dyes will respond to other frequencies. This phenomenon is known as cross-talk and is most prevalent when channels are scanned simultaneously. That is, when the 532nm wavelength is used, a response is seen from the cy5 dye and a smaller response is also seen from the cy3 dye. This means that two images extracted at the end of the process are not completely independent and carry slight readings from the other channel scanned as well. Images that are scanned sequentially do not necessarily overlap and need to be aligned in the image analysis step. A picture of the excitation is taken using a confocal microscope or CCD camera. The confocal lens has mirrors aligned at appropriate angles to ensure an accurate reading from the chip. Nevertheless, non-uniform readings occur resulting in slight warping of the array images and brighter readings in one region versus another. A photomultiplier tube (PMT) detects emission photons and converts them to electric current. Adjusting the voltage of the PMT influences the level of signal and noise intensity. Upping the PMT voltage alters the dynamic range of the signals and could result in saturation of high level pixels. PMT levels are in the control of microarray lab researchers and therefore vary dependent on facilities. Finally, the scan is digitized. The digitization process itself is only an approximation of the original image. As in any process involving the discretization of a continuous measure, the end product never completely represents the original measure.

Chapter 2

MICROARRAY IMAGE ANALYSIS

[1]

2.1 Current Methods

As the focus of this dissertation is signal extraction from spotted arrays, current techniques for signal extraction will be discussed. By far, most microarray labs use manually driven methods of image analysis using either the GenePix, ScanAlyze, or ImaGene packages. Current signal extraction procedures generally require the manual positioning of a grid over the arrayed image and the laborious visual inspection of each feature for quality. Accordingly, signal extraction currently can take upwards of two hours per array, with spot quality issues being highly subjective at best. In addition to being untimely, manual image analysis methods will differ based on the person performing them. If consistency in image analysis is desired, the method must be automated. Automated analysis that does not require human intervention and extracts signals at least as well as manual methods would already be a major advance in the way microarray images are analyzed.

Several semi-automated approaches have been developed to extract signals from spotted arrays. Rarely have these methods appeared in the literature, with some exceptions [9, 6]. Approaches vary in their level of subjectivity or their reliance on human input and borrow mostly from existing image analysis techniques. Current software packages either assist in the manual interpretation of array images or consist of automated processes to detect regions of interest or a little of both. Packages like GenePix specialize in assisting the user

[1] Sections of this chapter will appear as a journal article this year. The article is cited as: Bergemann, TL, Laws RJ, Quiaoit, F, and Zhao, LP. A Statistically Driven Approach for Image Segmentation and Signal Extraction in cDNA Microarrays (2004). *Journal of Computational Biology.* **11**: in press.

to specify the regions containing spots. Other packages, including Spot and BioDiscovery's AutoGene, use a mostly automated system for image processing. These packages borrow methods from the extensive image analysis literature and tailor them to microarray images, as will be elaborated upon in sections 1.2-1.4.

A representative analysis of array images involves four steps. Several techniques for these steps have been implemented in software packages such as GenePix and ImaGene. The first step is to align blocks and to position rows and columns of spots. The most common technique is to align blocks and grids manually. The second step is to detect spots. Spot detection methods vary widely from the fixed circles found by GenePix to a completely unstructured approach favored by Spot or Dapple [54, 7]. The third step is to estimate background intensity levels. Typically, background estimation involves taking the mean or median value of pixels locally surrounding each spot. This choice of background estimation, however, may be influenced by signal values and hence lead to under-estimation of signal values [54]. The fourth step is to compute signals, after accounting for background, and then integrate red and green channel values. Signal extraction involves taking means and medians for spots detected separately for each channel or from the union of the two spot detections. Channel values are integrated by taking the ratio.

There are numerous challenges to analyzing microarray image data. When performing grid alignment, manual adjustment is discouraged as it would introduce dubious subjective judgment into the procedure. More importantly, manual operation prohibits scaling-up image processing to handle many array experiments. A challenge associated with the spot detection stage is determining the level of restriction to be placed on spot structure. On one hand, making excessive assumptions about spot shapes may mislead estimation, while on the other hand, using non-parametric estimates with few assumptions may pick up artifact noise or background as well as signal regions. The challenge for background estimation is to carefully define what background consists of. Signal estimation challenges include ensuring the proper evaluation of signals in both channels and integrating these two channels

meaningfully. It is also important to quantify the variation of these signal values.

2.1.1 Notation

Consider sample array images (see Figures 2.1, 2.2) produced at the DNA Array Core of the Fred Hutchinson Cancer Research Center arraying facility. The arrays contain 6,144 clones representing segments of the yeast genome. cDNA arrays have spots grouped into blocks. Spots are arrayed similarly within each block. Let i denote the pixel position in the horizontal direction across the array image and j denote the pixel position in the vertical direction such that $r(i,j)$ is the red intensity value and $g(i,j)$ is the green intensity value of the ij^{th} pixel. Let S_{mn} denote a set of indices for the mn^{th} spot, $S_{mn} = \{(i,j)|(i,j)$ belongs to the mn^{th} spot$\}$, where $m = 1,\ldots,M$ designates the block row number and $n = 1,\ldots,N$ the block column number. Let B_{kl} denote a block of spots, $B_{kl} = \{S_{mn}, (m,n)$ are indices within kl^{th} block$\}$, where $k = 1,\ldots,K$ designates the row number and $l = 1,\ldots,L$ designates the column number.

The aim of microarray image segmentation is to find the pixels included in each block B_{kl} for all k,l, and each spot S_{mn} for all m,n. The goal of signal extraction is to quantify the intensity of each spot using the pixel values from that spot, $\{[r(i,j), g(i,j)] : (i,j) \in S_{mn} \in B_{kl}\}$. Further, the image analysis routines should estimate the local background values $\{[b^r(i,j), b^g(i,j)] : (i,j) \in S_{mn} \in B_{kl}\}$ for each channel, chosen to be a constant value for each spot.

Depending on the design of microarrays, several design parameters are known and useful for automated grid alignment. Specifically, the configuration of blocks is known, $K = 4$ by $L = 4$ in examples here. Within each block, the arrangement of rows and columns is also known, $M = 20$ by $N = 20$. The last row of each block is typically incomplete, i.e., $m(20) = 1,\ldots,4$. Let r_A be the expected radius for any spot. The size of a spot, dictated by the size of the pin tip, averages 15 pixels in diameter. The distance between any two adjacent spot centers is called the pitch, which will be denoted p. The pitch is based on the

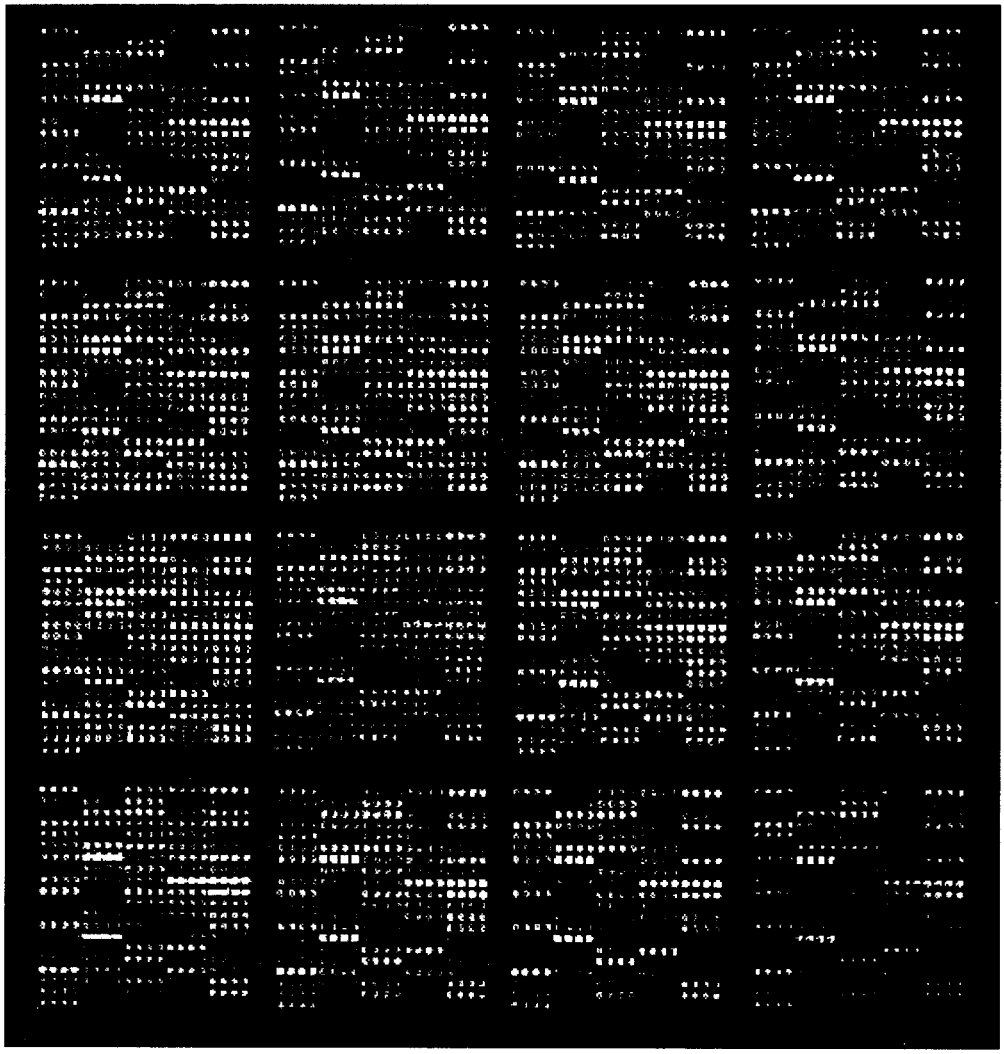


Figure 2.1: A quality image representing comparative hybridization of mRNA extracted from yeast cells.

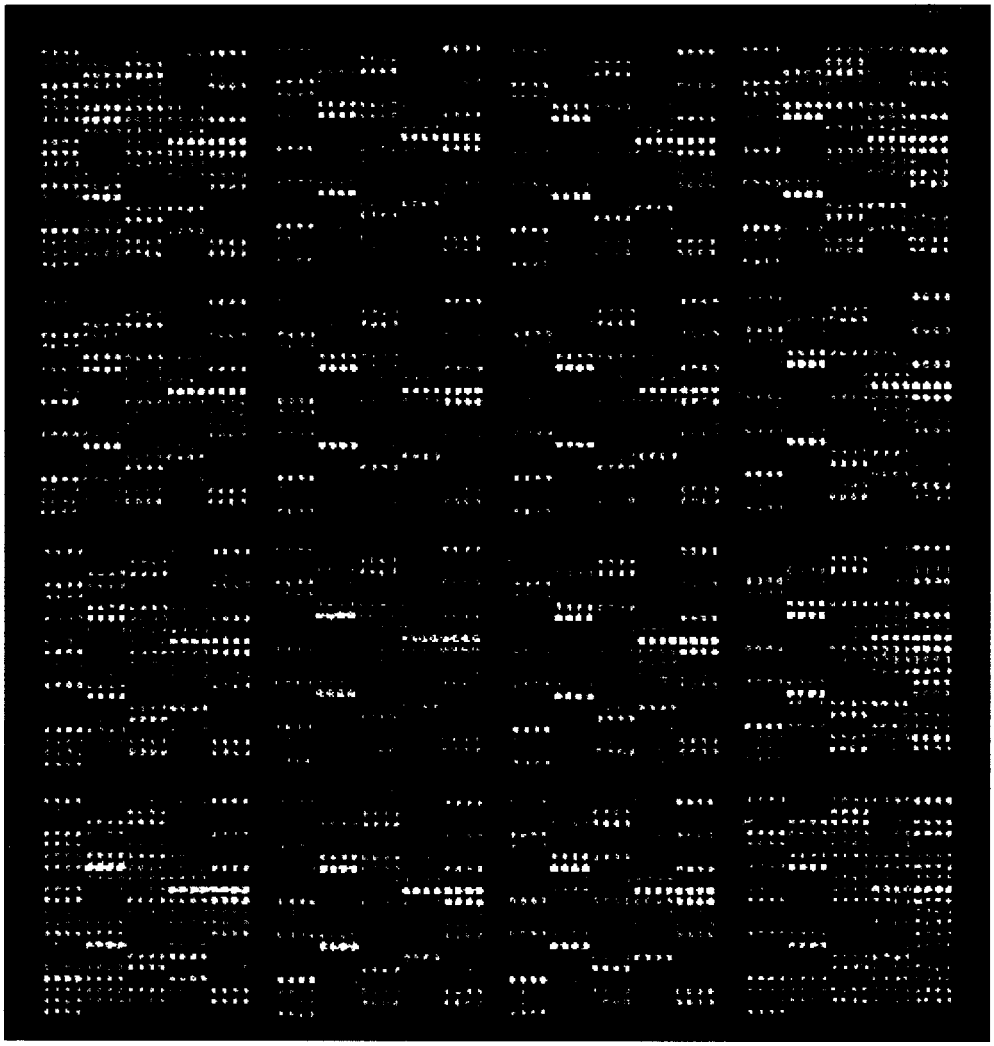


Figure 2.2: Image with low expression levels representing comparative hybridization of mRNA extracted from yeast cells.

placement of spots by the robot arm at roughly 200 microns apart. This distance converts to 20 pixels in these images. These parameters can be used as initial values, and later modified to fit alternate chip designs that arise.

Finally, this section presents an overall signal to background ratio where this ratio is defined by

$$SBR = \frac{1}{K \cdot L \cdot M \cdot N} \sum_{k,l} \sum_{S \in B_{k,l}} \frac{R_S + G_S}{b_S^r + b_S^g}$$

where the S subscript indicates a spot specific estimate, R_S, G_S refer to spot means within the red and green channels, and b_S^r and b_S^g refer to local background estimates within the red and green channels as described in Section 2.3.4. This ratio indicates the average differentiation between signal and background noise for a particular array. While the SBR is certainly not a definitive measure of array quality, it allows for a relative comparison between chips. That is, if an array has a larger SBR than another array image, the first will be more difficult to segment by image analysis packages than the second. This estimate to gives a general and heuristic indication of comparative array quality.

2.1.2 Grid Alignment

Very few published papers explicitly explain their grid alignment procedures. Thus, it is difficult to gauge what the current standard methods are for this task. As such, a brief review of most methods will be provided here and then more indepth analysis of one published grid alignment technique. For example, in Spot, they set up an initial grid template specified by the user which is then automatically adjusted for subsequent images [54]. The specifics of these adjustments are not provided. The same is true for the NHGRI microarray project [9]. Of note, is the body of work written by an Austrian group that outline their alignment procedures in detail [5]. Although their methods use many rigorous steps and were developed for nylon filter arrays, they are worth discussing as the methods are common in signal processing. It can not be said how reliable the following methods are for array images as no rigorous evaluation of them was provided in the literature.

Alignment begins by filtering to enlarge the signal to noise ratio. This "matched filter" requires the selection of sample spots by the user. These sample patches are compared to the original image via a dot product of sorts. Suppose there exist image patches that are vectorized, $\vec{f}_i, i = 1, \ldots, K$ and of length D. Patches are averaged and standardized,

$$\vec{m} = \frac{1}{F}\sum_{i=1}^{F}(\vec{f}_i - \hat{\mu}_i) \text{ where } \hat{\mu}_i = \frac{1}{D}\sum_{j=1}^{D} f_{ij}.$$

The resulting matched filter image is $R^M = (\vec{z} - \bar{z}) \cdot \vec{m}$ where $\vec{z}$ is a vectorized version of a sample image also of length D, and $\bar{z}$ is the average of $\vec{z}$. R^M will ideally have brighter spot regions.

Slight rotations in the array image may be accounted for using various projections from the two dimensional image to marginal vectors. Brändle et al used a radon transform to project their matched filter images, R^M, into one dimension along direction θ [5]. If $\theta = 0$, the image is projected onto the x-axis and if $\theta = \pi/2$, the image is projected onto the y-axis. If no rotation is required, a projection onto the x- or y-axis will result in a vector with clear peaks corresponding to columns or rows of spots. If the image has been slightly rotated, the projections will not exhibit peak patterns and must therefore be adjusted. The authors examined projections for deviations in θ of $0.125°$ for the axes and chose θ such that peak patterns were maximal.

Their procedure allows for two possible grid alignment methods. The first method looks for maximum local values in a matched filter image R^M. The second method uses the marginal projections from a transformed image. The position of peaks in these projections designate row and column position [5]. After implementing one of these two methods, the next step is to confirm the grid estimates and eliminate false rows or columns. To this end they employ a parametric model. It seems that for spotted arrays printed on glass slides this last step would be unnecessary since the signal to noise ratio is higher than for nylon filters.

Evaluating the above outlined method is difficult without any published results. Never-

theless, in this author's experience, their approach seems reasonable and representative of intuition. There are likely to be several filters that would eliminate image noise equally as well. Rotations in the images could also be accounted for using any of a number of other projections other than the radon transform. Grid alignment using peak identification was the favored approach in the published literature. Note that these methods do not account for jumps in grids within a subsection of a block. Vast jumps greater than pitch distance can not reasonably be detected by any current method.

2.1.3 Spot Detection

There are many spot detection methods for microarrays, both published and unpublished. Therefore, only concepts in image segmentation will be provided while avoiding an exhaustive review. Automated methods to determine spot placement (image segmentation) include thresholding, estimation of two-dimensional gradients, region growing, and model-based estimates.

Thresholding

Thresholding involves choosing an optimal cut-off point such that all pixel intensity values above that point are designated as signal intensities and all pixel intensities below the cut-off are designated as background. Commonly, thresholds are chosen by finding the point that best separates two points in a bimodal distribution as estimated by a histogram [3, 29]. In the microarray arena, thresholding would be performed within each grid alignment frame. For spots of variable intensity, however, pixels of low intensity within a spot region are not likely to fall above the threshold and would therefore be categorized as background. Also artifact noise of high intensity will be categorized as part of the region of interest. Figure 2.3 gives a sample segmentation. Attempts to avoid image segmentation via thresholding would involve estimation of a bimodal distribution to extract values for both signal intensity and background. Complicating these attempts are the difficulties inherent in estimating a

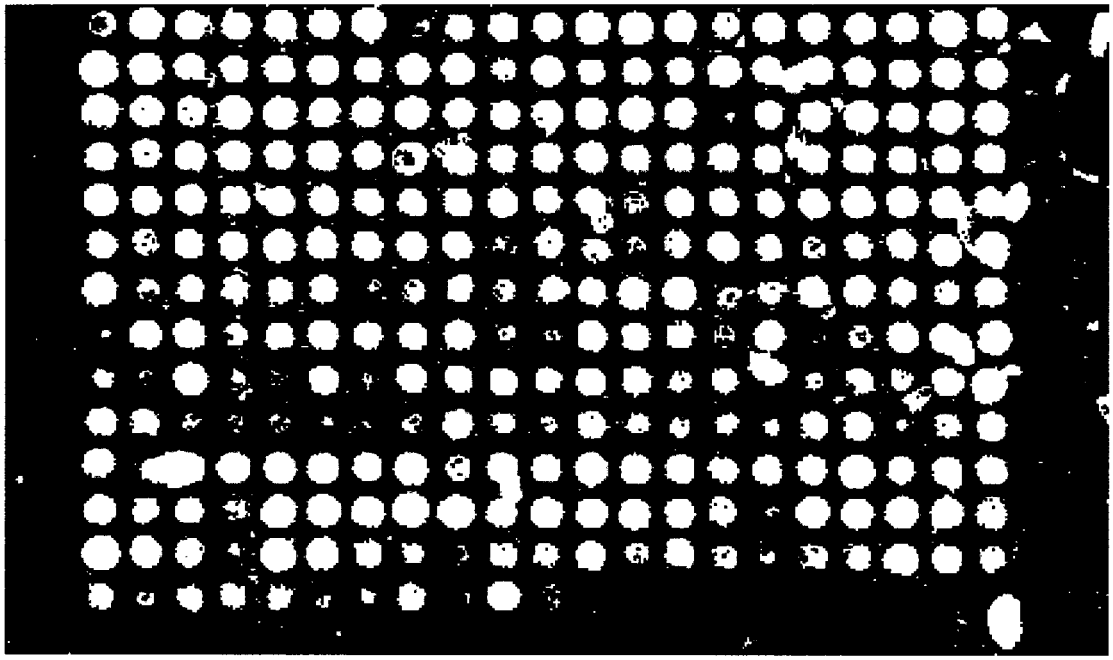


Figure 2.3: An example to illustrate image segmentation using thresholding.

bimodal distribution. Recent attempts at this problem can be quite complex involving MCMC algorithms [52]. While this approach is likely to be useful on a small scale, the computing time would become exhaustive when trying to estimate at least 6,000 bimodal distributions per microarray slide.

Gradient Maximization

Alternatively, spots can be identified by maximizing gradients in both directions. Dapple uses Laplacian of Gaussian edge detection within each grid frame [7]. Prewitt's gradient filter assumes a 3 by 3 pixel planar surface that is estimated via least squares and uses this surface to compute the maximal gradient [3].

The often cited Canny method uses gradient maximization followed by adaptive thresh-

olding [8]. It is shown that marking edges at gradient maxima is equivalent to finding zero-crossings of a nonlinear differential operator. Three criteria are used to find an optimal edge detector: (1) optimize the signal to noise ratio (2) best localization of the edge (3) elimination of multiple edges at one locality. The paper shows that the first derivative of a Gaussian is a nearly optimal filter for these three conditions and has the added advantage of being well defined. Gaussian filters are also used in Laplacian of Gaussian edge detection. The image is convolved with a Gaussian filter and then edge detection is performed using thresholds. Two thresholds are set, strong and weak, such that pixels within a ridge above the strong threshold are tracked along the ridge until falling below the weak threshold. In this way, only those areas above the weak threshold that are also contiguous to a region above the strong threshold are included in the final image segmentation.

Gradient methods do not make assumptions about the structure of the image. Hence, for microarray data, these techniques often segment multiple regions within one spot and edges lack connectivity. These methods also do not differentiate well between noise of high intensity and actual signal.

Seeded Region Growing

Seeded region growing seems best suited to the structure of microarray images [2]. Seeds, or pixels, are chosen as initial values for each region of interest. The most intuitive place to set seed points in microarray images are the grid alignment centers. The algorithm then "grows" from each seed point and tests whether surrounding pixels should be added using an arbitrary distance metric, typically Euclidean. The algorithm has been applied to array data in a package called Spot [54]. It seems that seeded region growing could be a feasible analysis method if thresholds were set such that low intensity spots did not grow beyond a typical size or take on what are clearly irregular shapes.

Seeded region growing seems best suited to the structure of microarray images [2]. Seeds, or pixels, are chosen as initial values for each region of interest. The most intuitive place

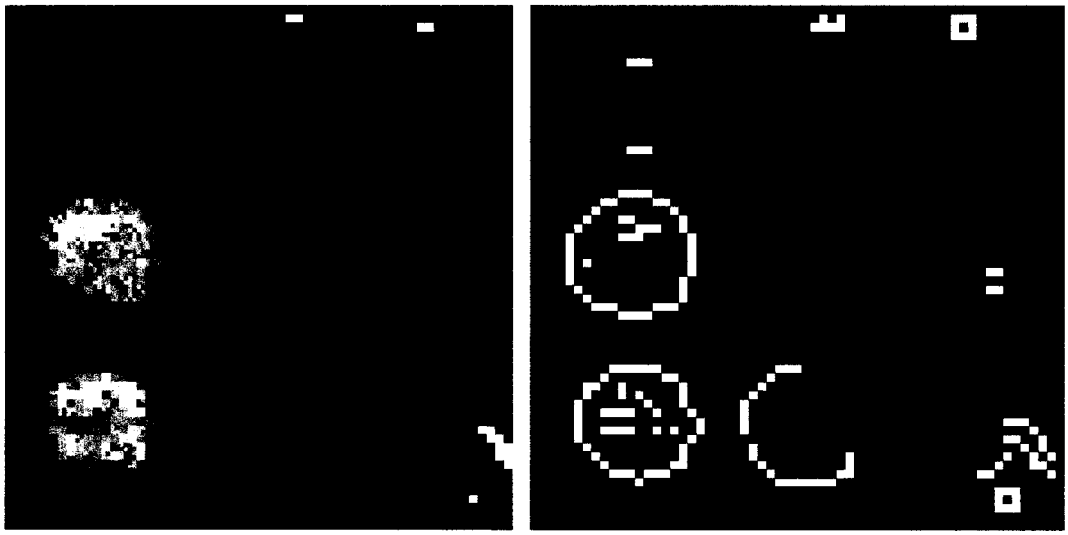


Figure 2.4: An example to illustrate image segmentation using Prewitt's method.

to set seed points in microarray images are the grid alignment centers. The algorithm then "grows" from each seed point and tests whether surrounding pixels should be added using an arbitrary distance metric, typically Euclidean. The algorithm has been applied to microarray data in a package called Spot [54]. For array data, spots of low intensity value tend to "grow" too large and take on irregular shapes, including what is undoubtedly background in the signal region. Figure 2.5 represents seeded region growing after smoothing a fourth root transformation of the original image. It seems that seeded region growing could be a feasible analysis method if thresholds were set such that low intensity spots did not grow beyond a typical size or take on what are clearly irregular shapes.

Model-based

Model-based image segmentation assumes that the areas of interest follow a parametric form. The most simple model-based method for array data assumes that all spots form circles. Then using a loss function criterion such as least squares, circles of optimal size are placed

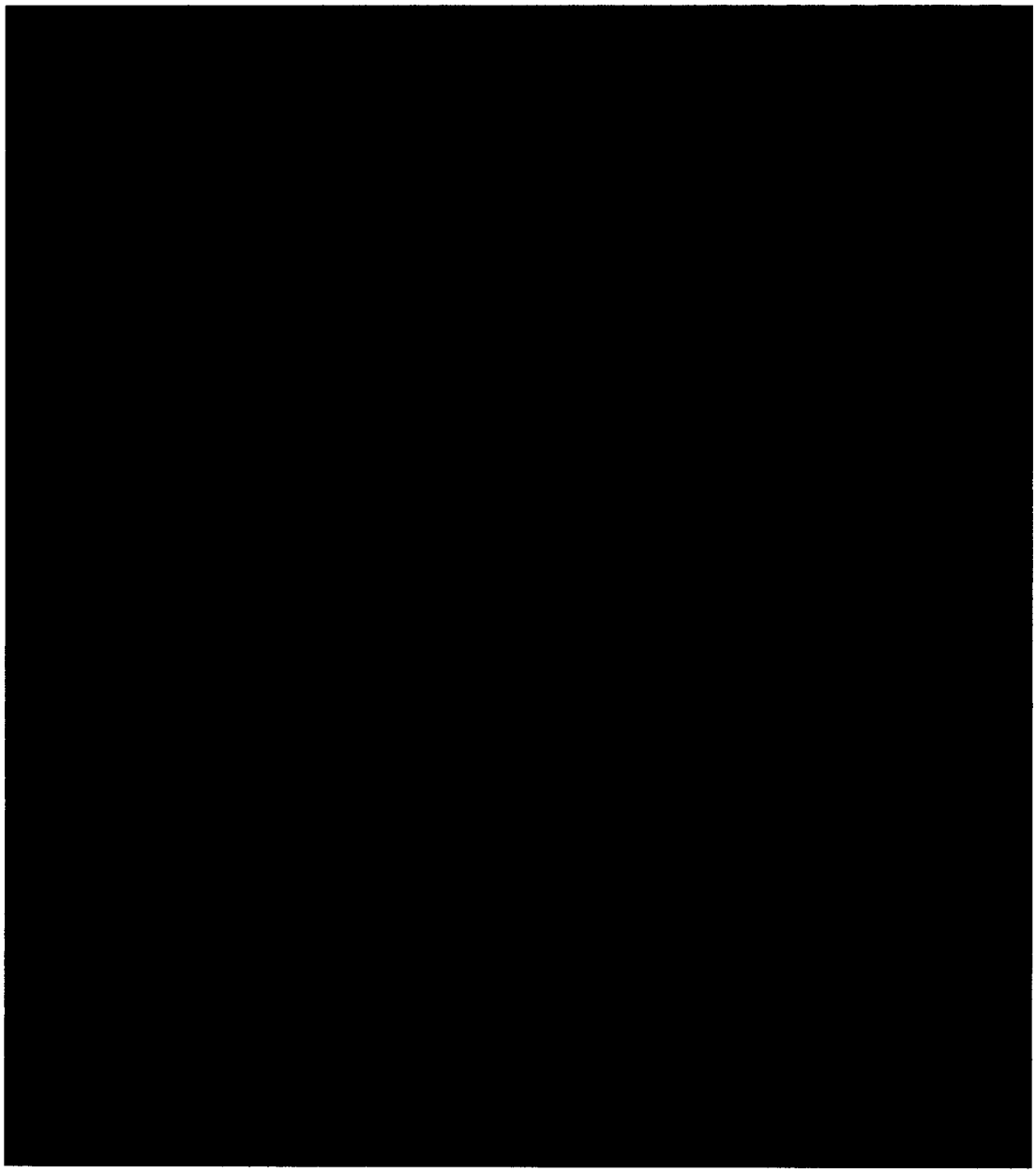


Figure 2.5: An example to illustrate image segmentation using seeded region growing.

in their optimal positions to approximate the spot area. GenePix uses this approach. More complex model-based methods might assume that the array data has some sort of multi-modal distribution in two dimensions. MCMC algorithms can be used to estimate the form of this distribution. Another model-based spot detection method assumes that each spot follows a Gaussian distribution [5]. In particular, this method looks to estimate parameters for spot intensity, spot size, and spot location. The model is defined as

$$z(\vec{p}) = a \exp[-\frac{1}{2}(\vec{p} - \vec{\mu})' \Sigma^{-1} (\vec{p} - \vec{\mu})]$$

where a =spot intensity, μ =spot location, and Σ =spot size. Note that it is feasible to combine model-based segmentation with other techniques such as thresholding or region growing.

2.1.4 Background estimation

Background estimation techniques for microarrays fall into three categories: local correction, global correction, and negative controls. The issues involved with each will be further discussed in Section 2.2.3. The most commonly adopted correction is a local adjustment and is the default in the majority of image analysis software packages. Negative controls are most prominently used by Agilent Technologies as part of their design for ink-jet technology arrays [15]. Some groups such as Applied Precision prefer global background corrections [6].

Background noise can be estimated or accounted for by using a filter. The simplest filters are linear, but there are also non-linear and morphological filters [3]. A simple non-linear filter is median filtering where for each pixel in an image, a window about that pixel is chosen. The window consists of an n-by-n region surrounding the pixel. The background estimate for that pixel is set to the median value of the window.

More complex non-linear background estimates include morphological filters [45]. In its simplest form, the filter works as follows. Let y_{ij} represent the original image. Suppose

$\sqrt{k^2+l^2} \leq R$ for integers k, l and some arbitrary R. The filtered image is

$$z_{ij} = \max_{k,l} x_{i+k,j+l} \text{ where } x_{ij} = \min_{k,l} y_{i+k,j+l}.$$

This represents a morphological opening, which is used by Spot [54]. Reversing the minimum and maximum would result in a morphological closing.

Most microarray researchers adjust for background noise by selecting a window around each spot and calculating the mean or median of that window. In skewed distributions, summary measures such as the mean and median are known to overestimate the central tendency of the population. An obvious extension of median filtering is to use some other percentile, such as the 25th percentile of the window. Background noise can also be found by estimating the density of local background regions. The NIH microarray group uses histograms to measure this density. The mode is selected as the background estimate and the left tail of the histogram is used to estimate the standard deviation of this estimate [9].

Some image analysis packages use grid alignment to identify areas likely to be outside of the spot regions. Namely the corners of each grid alignment frame of size p^2 are designated as pockets of background. A summary variable for each of the four pockets can be used to assign a local background estimate. Another group uses median filters to estimate typical background noise but density estimators for atypical background [21]. Robust regression methods are used to approximate the background by a Gaussian distribution. Separate fits are assigned to the right and left tails of the normal. Estimates of μ and σ^+, the variance of the right tail, are used to construct confidence intervals. These confidence intervals test for the presence of non-homogeneous noise due to artifacts. Spot regions with significant artifact noise are flagged.

Some researchers choose to adjust for global background, rather than make a local adjustment. The global estimate can also be calculated in various ways. Typically the regions outside of the array blocks are used for this global calculation. Or the global estimate can be an average of all local estimates. Goryachev et al provide a happy medium

between global and local background [21]. They use a neighborhood of roughly 5 by 5 spots to adjust for background. Optimal neighborhood size was determined by examining spatial correlations via a Fourier analysis.

Finally, some investigators are opting not to adjust for background at all. This is similar in spirit to global corrections.

Correction for background remains controversial at this time. The controversy derives from the fact that different background corrections can result in vastly different estimates [54]. At issue is whether the correction should be global or local, and what the probable measure of this background should be. One way to resolve this controversy is to better understand the hybridization process in microarray experiments and to identify sources of background variations inherent in microarray images.

2.1.5 Signal Extraction and Integrating Channels

Most researchers create a summary variable to represent expression values for each gene within each channel of an experiment. The summary variable, usually the mean or median, uses all pixels deemed to be within the spot region. A local background estimate is then subtracted from this summary variable. Alternatives are available. As explained earlier, attempts can be made to estimate a bimodal distribution and glean both signal and background from this estimate. It has also been suggested by several groups that a linear regression be estimated for all pixels within a grid frame. In this way the regression line fit to the red versus green channels, grounded by background values, gives a slope estimate equivalent to a signal estimate. This regression approach combines signal estimation, background estimation, and channel integration.

Boundary pixels create problems for any signal extraction procedure. The boundary generally consists of pixels that are influenced both by signal and background, therefore belonging to both regions and neither. Little of the published research comments on their methods for dealing with boundary pixels. GenePix forms a circle slightly larger than the

spot and eliminates those pixels both from background and signal calculations.

Image segmentation is usually performed separately for each channel. Combining the two images into one spot can be done in several ways. Spot takes the union of the segmentation regions for each channel [54]. Dapple uses the average of the two channel segmentations if both detections are designated at the same quality level. If the spot in one channel is a higher quality category, then the detection for this spot is used for both channels [7].

The current standard to compare red and green channels on microarray images is to calculate the log-ratio of intensities for each gene. The ratio is used because cDNA concentration cannot be precisely measured for each assay and hence must be controlled for by using a relative estimate. The standard approach of integrating red and green channels is to compute the logarithmic ratio of two intensity values, that is,

$$\log_2 \left\{ \sum_{(i,j)\in S} [r(i,j) - b^r(i,j)] \Big/ \sum_{(i,j)\in S} [g(i,j) - b^g(i,j)] \right\},$$

where all quantities are evaluated using chosen image analysis algorithms. The above logarithmic ratios are commonly extracted and are used in practically all of array data analysis. Practical applications using the above estimate indicate that when intensity levels in both channels are large, that the estimated ratio is reproducible. Otherwise, the reproducibility is quite low, partially because a small change to a low intensity value could have a substantial impact on the ratio estimate. This is further discussed in Section 2.2.4.

2.2 Challenges and Issues

The largest challenge to microarray image analysis is the highly competitive nature of the field. Nearly all available software is proprietary even when developed at academic institutions. This forces scientists to reinvent the wheel if they wish to make improvements and impedes the advancement of the field. To overcome this impediment, groups must be encouraged to publish analytic methods and therefore make them explicit. Proprietary

software should not only show the results of the packaged algorithms but also publish the reasons for the reliability of their software. Justifying reliability requires the delineations of the algorithms used. Ultimately and ideally open source packages will exist that can be built upon and improved.

Another challenge to data extraction from array images are the biases and variability present in the data. This noise is the result of both technological and experimental components. Desirable image analysis and data extraction will attempt to account for these measurement errors.

2.2.1 Grid Alignment

Alignment of microarray grids involves setting up an initial target from which to search for each spot on a microarray image. Because spots are printed onto microarrays in a systematic fashion, starting an image analysis routine by setting up basic grids is intuitive. Grid alignment involves identifying each block or quadrant, created by its own pin tip, on the image. For each block, rows and columns must be drawn such that they overlap the spots within. When each spot region has been targeted by grid alignment, spot detection ensues. Although some researchers regard grid alignment as a trivial exercise, practical issues exist to challenge such a view. Three examples of such problems follow in Figure 2.6.

1. Some labs have difficulty printing spots parallel to the edge of the array slide. Or during the image scan, the slide may be slightly askew. This results in a mild rotation in the final image. Hence some grids may have to be rotated two or three degrees before lines can be drawn over them [54].

2. Because of anomalies in the printing process, grids will often shift on the slide. Spots are rarely equally spaced within the grid. Sometimes egregious errors occur in the printing process and a region may jump a distance of a few rows or columns [38].

3. Arrays with large amount of background noise relative to signal will be more trou-

blesome. High levels of background noise due to dust or water marks might fool an automated system into placing grids over the background noise. Background varying grossly over the array will also create problems.

Gross artifact noise on array chips will impede grid alignment even for the most robust of methods. Analysis packages would benefit from a flagging mechanism to assess situations where alignment is unlikely to succeed. For example, if an error in printing results in an entire row jump for some subset of the row's spots, most alignment packages can not accommodate this jump. Hence, a flag would be useful.

Most available software packages have yet to make grid alignment a fast, fully automated procedure. Goals include performing grid alignment for an entire array in less than a minute on a run of the mill PC. Fully automated can be defined as methods requiring no human intervention and only a few set parameters such as expected number of rows M, columns N, blocks $K \times L$, spot size r, and spot distance p.

2.2.2 Spot Detection

Many factors can influence decent spot detection including presence of artifact noise, irregular spot shapes, and low levels of mRNA hybridization. Here too, flagging procedures to help identify poor detections would be desired. In particular, it is challenging to be able to determine the difference between particles with high intensity values and spots. Even when evaluating an array image by the human eye, this determination can be debatable. Finding objective measures by which to differentiate signal from high artifact noise remains an open problem. Until these objective measures can be applied, manual manipulation of a software package's detection should be permissible. Again, spot detection should be fully automated with a list of parameters as input and capable of segmentation inside of five minutes (100 images in an overnight batch process).

Genes with low levels of transcription for a given time and tissue type will result in array spot intensities often indistinguishable from background. In fact, some spots will

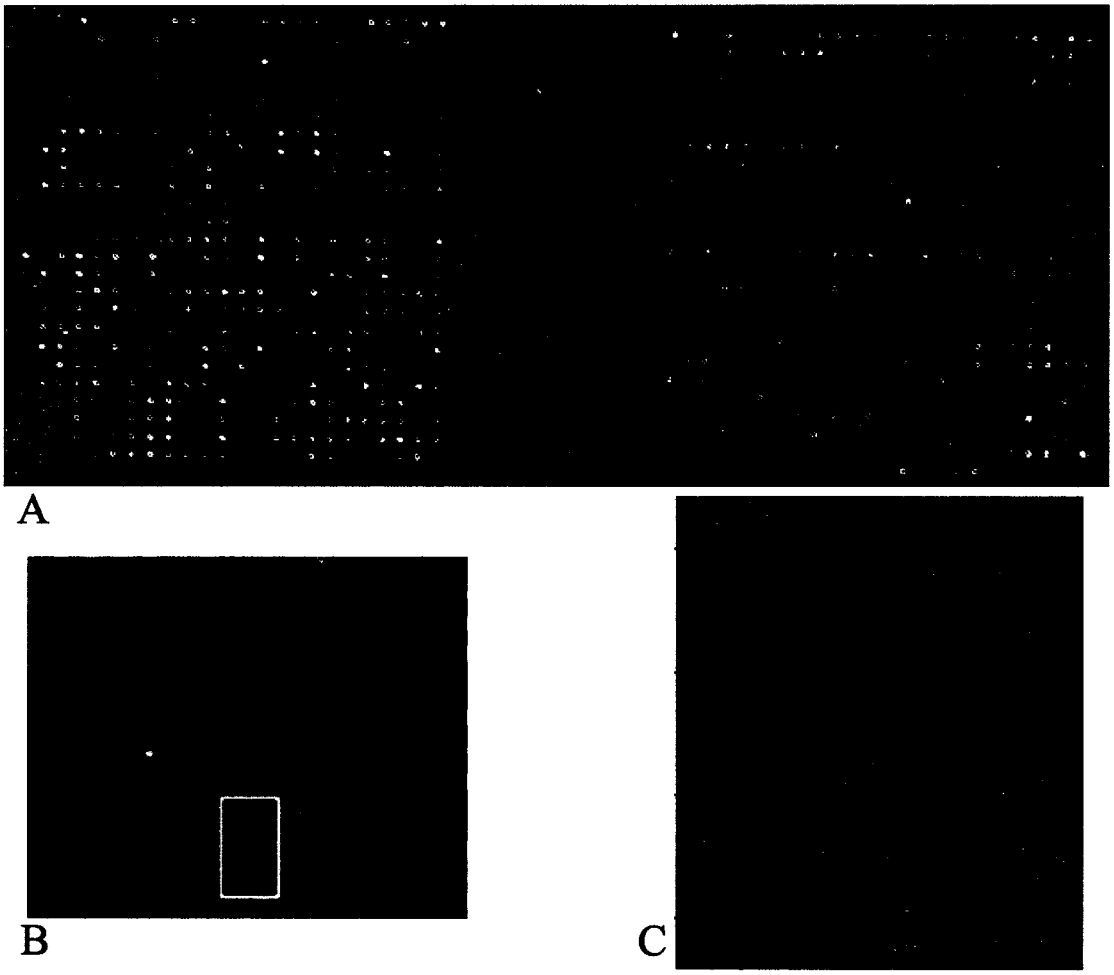


Figure 2.6: Examples of array images with faulty grid alignment. Example A shows problems with rotation, example B shows grid jumping, and example C displays an array with large amounts of background noise with high pixel intensity values.

have expression values lower than surrounding background. Researchers will throw out genes with expression values below certain arbitrary thresholds. The resulting missing data is not missing at random as the data is missing due to low expression levels. Hence, throwing out these "missing" values biases the final results. Image analysis packages therefore need to be able to correctly identify spots as having low expression values and assigning values of zero instead of missing. After all, a value of zero is informative about the activity of a particular gene for a given condition.

2.2.3 Background Estimation

The largest issue in background estimation is answering the following question: What is background? For some say that negative controls represent background, others say noise, and others yet say the inherent excitation of the glass slide. In general, the goal is to adjust spot intensity values for biasing influences resulting from the technology. The way in which intensity values are adjusted will be dependent on what people think to be a bias.

Negative controls are clones lacking complementary mRNA transcripts in the tissue condition of interest. These controls give evidence for what happens to a spot that theoretically has zero intensity. Some labs spike in exogenous mRNA from another species as a negative control. These controls only account for variation after the time of the spiking. In addition, if the exogenous sample is homologous to areas of the genome being tested, there might be non-specific hybridization to the control clones greater than the average cross-hybridization rates. In practice, such genes that consistently express zero transcript do not exist. And, if they did exist, the gene clones must be placed in spatially appropriate ways. That is, the environment of the array varies from place to place. So, a clone placed on one part of the array can not be adjusted for with a control placed on another part of the array. An open question is to determine the optimal placement of negative controls to account for spatial variability within an array experiment.

Local background estimates provide an idea of the noise surrounding each individual

spot. Hence it is possible to measure the natural intensity of the glass slide for each channel at all places in an experiment. Nevertheless, the condition of the clone areas and the unprinted areas could be quite different. And so adjusting a clone area by an unprinted area might be unreasonable. Local areas can also be contaminated by artifacts such as dust and water marks therefore biasing the background estimates. Local background regions can also be contaminated by the signal itself if the region is particularly large or bright. Finally, as discussed in section 1.4, debate continues about the best local estimate.

Global background corrections give an idea of baseline noise on the entire array. They are also robust to spatial anomalies that might influence local estimates. For example, when signal intensities are lower than the natural autofluorescence of the surrounding glass slide, local corrections will result in more negative values. As negative values are nonsensical, it is better to correct for noise in a way that still results in meaningful signal interpretation. These corrections, however, ignore spatial variability in background noise.

2.2.4 Integrating Red and Green Channels

The standard for comparing expression values between two tissue types is to use the log ratio. Few alternatives have presented themselves. The log ratio is favored because of its easy interpretability and intuitiveness. Any alternative must also have these qualities.

Ratios are inherently unstable because low values in the denominator result in essentially infinite estimates. Thus, difficulties arise when attempting to quantify genes with low levels of expression. Further, if local background estimates exceed signal estimates, ratio estimates will be negative and hence log-ratio estimates do not exist. The theoretical considerations when examining ratios tend to be tricky as well. For example, if the red intensities are normally distributed and the green intensities are normally distributed, then the ratio of the two follows a Cauchy distribution. Since the Cauchy distribution has infinite moments, it becomes problematic to quantify variance, i.e. reliability of the ratio. Lastly, ratios do not account for the competitive nature of the hybridization procedure on the microarray.

That is, once a probe cDNA strand has a hybridized to a target cDNA strand labeled red, that probe strand cannot subsequently hybridize to a green labeled strand.

When integrating the two channels of an array experiment, expression estimates must often be adjusted for by differences in labeling and scanning efficiencies. This is referred to as normalization and is a much debated topic (not a rare thing in the microarray world). A plethora of papers have been written on the subject [21, 55]. But the subject falls outside the scope of this dissertation and will not be discussed further here.

2.3 SignalViewer: An analytic tool for microarray images

Here, as a response to issues raised in 2.2, a software application in development is presented. The package was developed in MATLAB (http://www.mathworks.com) and is by no means intended as the final answer in image analysis for microarrays. Rather, it is preferred that the package can be built on by other scientists as well to continually further this endeavor. SignalViewer is operated independent of MATLAB, however, as a stand-alone application.

2.3.1 Basic infrastructure

The software reads image files in TIFF format. The data can be imported either as two separate images (red and green channels) or a combined file where the user specifies the order of the channels. The software also reads a text file with gene names and identifiers as well as layout parameters for the images. The specified layout parameters necessary are expected number of rows M, columns N, blocks $K \times L$, spot size r, and spot distance p. The layout parameters can also be specified manually in a layout parameter window. At any point in the process information about the current image analysis can be saved and opened later.

Analysis results from the segmented images can be saved and/or exported to a text file. The exported text file contains gene, spot, and block identifiers in addition to spot flags, mean and median spot intensities, standard error estimates of spot intensity, and

local background estimates for each channel.

SignalViewer allows the user to zoom into microarray images at different levels using GUIs. Images can be viewed at the array level, block level, and individual spot level as shown in Figure 2.7. There is also a general zoom feature. Different levels can be viewed with or without resulting grid alignment and image segmentation.

Three plotting features are available to the user at any level. The plots represent the data in various ways:

- A scatterplot graphs pixel intensities from the green channel versus the red channel in two dimensions.

- A histogram gives a density estimate of the area of interest for each channel, red bars for the red channel and green bars for the green channel.

- A 3D plot shows the area of interest in three dimensions with two viewing options, either a colorbar indicating pixel intensities or a relief map in gray-scale where the altitude of a point in the plot represents the intensity of the pixel.

Grid alignment, spot detection, background estimation and flagging are fully automated procedures in the software. Running a sample image of 6,144 spots on a Compaq AMD Athlon 697 MHz processor took 73 seconds to load the images, analyze them, and export the text file. Flags are indicated by SignalViewer as red circles in default positions. If these automated procedures fail or are not suitable to the user, a manual override is available both for block and spot placement. Spots that have been manually drawn have green circles about them instead of the usual white. The export file indicates that the spots have been manually drawn.

Constructing simulated images is work in progress. Currently, it is an option within the software allowing the user to specify standard distributions for spot intensities and background noise. This allows for a cursory examination of the ability of the package to correctly estimate appropriate signal intensities.

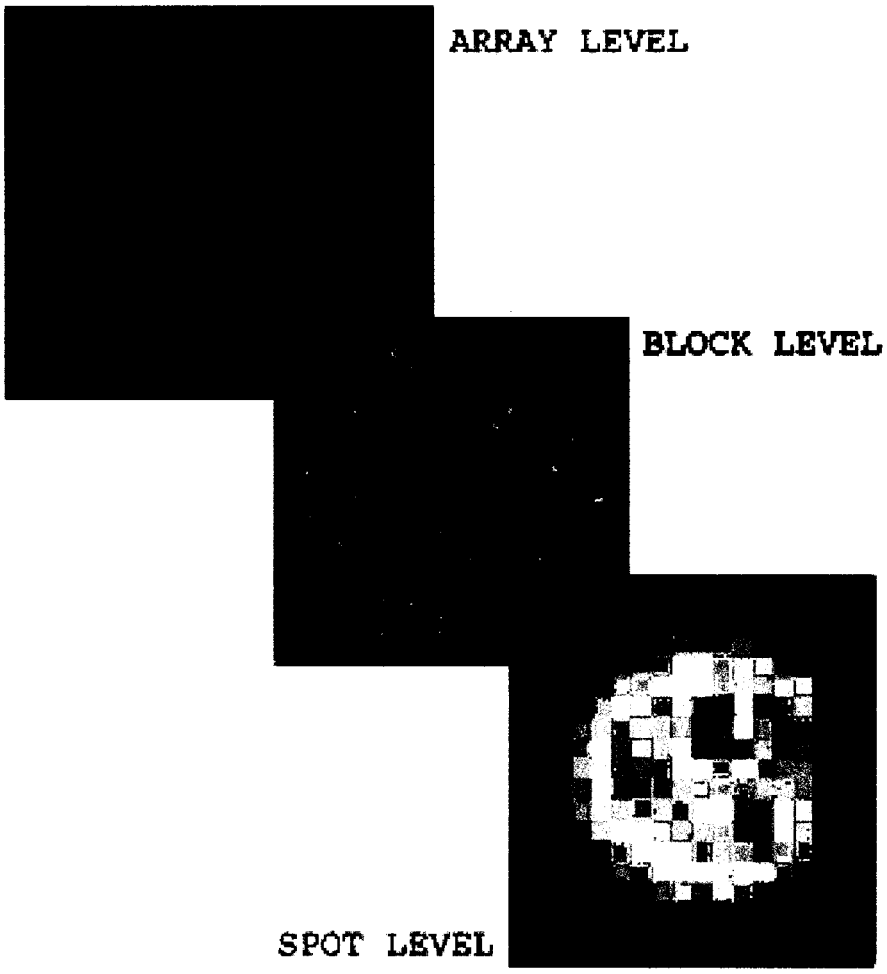


Figure 2.7: Interactive levels of viewing built into SignalViewer focusing on three major steps in image analysis: block identification, grid alignment, and spot detection.

2.3.2 Grid alignment

Grid alignment involves identification of blocks and placement of rows and columns within each block. Figure 2.8 diagrams the SignalViewer algorithm. Block identification is an iterative algorithm using the placement of previous blocks to assist in the search for the next. The upper left hand corner of the first block is found by projecting onto two axes and thresholding. For $z(i,j)$ generically representing the intensity at pixel (i,j) the axis projections are

$$\begin{aligned} x(i) &= \{\mathcal{F}^{-1}(0.65) \text{ where } z(i,j) \in \mathcal{F} : i \leq p * M\} \text{ and} \\ y(j) &= \{\mathcal{F}^{-1}(0.65) \text{ where } z[(i,j)] : j \leq p * N\}. \end{aligned}$$

A threshold is set as the 62nd percentile of these projections and the first point crossing this threshold is the upper left hand corner. Thus the corner is $(\min\{i : x(i) > c_x, \forall i\}, \min\{j : y(j) > c_y, \forall j\})$ where c_x, c_y is the 62^{nd} percentile of $x(i)$ and $x(j)$ respectively. The primary reason for choosing percentiles is to minimize the influence of outliers without losing efficiency. The level of the percentiles are roughly based on the relative area of signal to background. Admittedly, the choice of 62^{nd} and 65^{th} percentiles is still arbitrary. This method is known to be functional though on several images from three different lab platforms.

After the first block is identified, the next iteration of projection and thresholding is performed to find the blocks to the right and below. The searching space is a block sized area starting where the previous block leaves off. The area of the block is estimated from the expected pitch and number of spots.

Once blocks are found, grid alignment is performed within each block. Again projections onto the x- and y- axes are 65th percentiles. A loess smooth is performed on these projections using a spanning window of 3% of the total data [12]. A numerical first derivate of the loess smooth is used to detect peaks in the projections. These peaks are then used to draw rows

and columns.

Two automated corrections are in place to deal with faulty block placement and grid alignment. First, if a block is placed off the image region it is bumped back to the image border. Second, if the peaks detected during grid alignment are too far apart, a default grid is used instead. Peaks that are too far apart are defined by large deviation from the estimated distance between spots, p.

2.3.3 Spot detection

Grid alignment is used as a point reference for each spot in a block. Around each grid center point, a frame of area p^2 is drawn. This frame provides the data used for individual spot detection. Again, projections are taken and smoothing and thresholding are performed. The projections onto the x-axis(y-axis) are sums over each column(row).

$$x(i) = \sum_j \{[z(i,j)] : (i,j) \in F\} \text{ and } y(j) = \sum_i \{z[(i,j)] : (i,j) \in F\}$$

for grid alignment frame F. Smoothing of these projections uses a loess with a span window of 25%. The threshold is set as the midpoint in the range of the data.

The first points to cross the thresholds are used to define the height and width of an ellipse, a and b. Figure 2.9 illustrates this. These values are used to construct a digital ellipse.

$$(i,j) \in S \text{ if } \frac{(i-p/2)^2}{a^2} + \frac{(j-p/2)^2}{b^2} \leq 1$$

Pixels within the ellipse and used to estimate the intensity for that spot.

2.3.4 Background estimation

As pointed out in Section 2.2.3, there remains many questions as to the best way to estimate background. Given current understanding of the technology, there is probably no single best interpretation. To ensure generality, the author proposes using the neighborhood of each

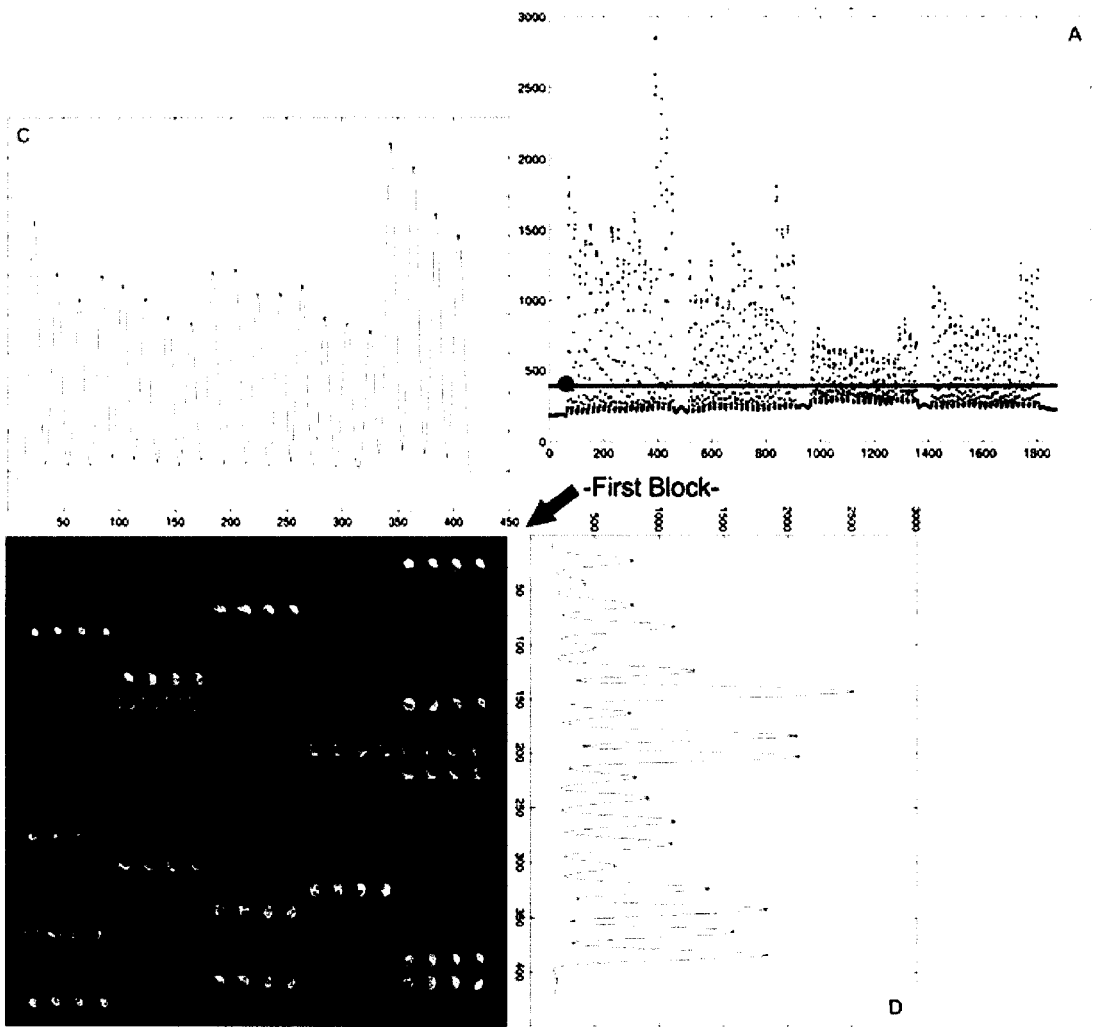


Figure 2.8: A: The projection of an array image onto the x-axis. The black dot indicates where the first block will begin. B: The image of the first block. C: The projection of B onto the x-axis. A loess smooth is performed and peaks identified. D: The projection of B onto the y-axis.

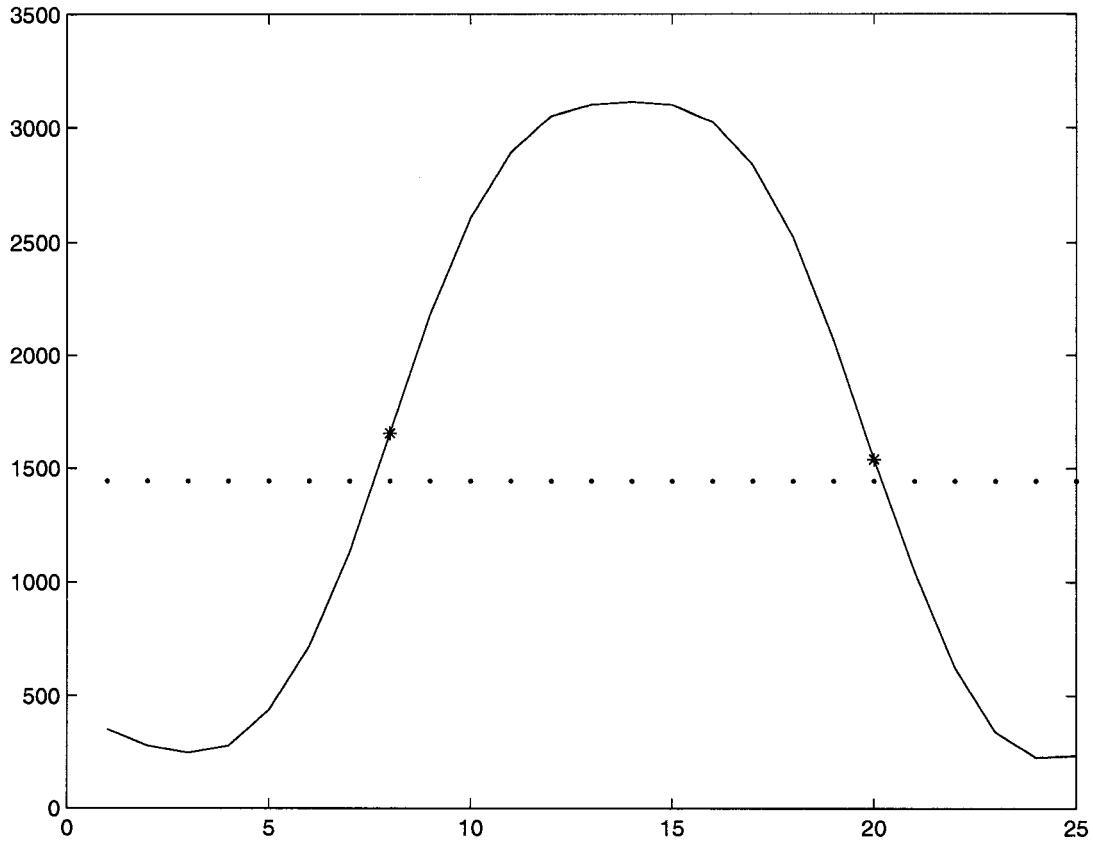


Figure 2.9: Cross-sectional used for spot detection. The projection represents a sum taken over pixel rows within a grid alignment frame. The projection is smoothed via a loess. Pixels crossing a threshold are marked as endpoints for the ellipse y-axis.

spot to estimate local background within each channel on the array. For each grid alignment frame a histogram is constructed using all the pixels within the frame. The most common intensity value is chosen as the background estimate. That is,

$$b^r = \{k : n_k > n_{k'} \ \forall k, k' \in 1, \cdots, 2^{16} \text{ where } n_k = \sum_{i,j} I(r(i,j) = k)\}$$

estimates background for the red channel and similarly for b^g. Because the intensity distributions in local regions are severely skewed, the mode is chosen as it is a more accurate estimate of central tendency than either the mean or median. Several possible neighborhoods could be used to designate the background area. These include local immediate regions around a spot, local including several adjoining spots, or global including the entire array. An immediate local neighborhood is chosen to estimate the mode. Further discussion on background estimation and choice of neighborhood is in Section 2.1.4.

Estimating the mode, rather than the mean or median, is associated with some drawbacks. The largest hurdle is how to quantify the variance of the mode estimate. Asymptotic theory has been developed to construct closed form estimates for confidence intervals relying on slow convergence to the Chernoff distribution [40]. Calculations of these confidence intervals rely, however, upon numerical estimates of the second derivative of the probability distribution function. Because no distributional form for background estimation is assumed, but rather estimated empirically, this makes calculation of the second derivative more difficult. Silverman suggests methods for estimation of the second derivative using Taylor series approximation, but the method is rather ad hoc [44]. Instead, SignalViewer uses the spread of the data as an empirical variance estimate. This is equivalent to σ^- used by the NIH [9].

2.3.5 Flagging

Four automated corrections are in place to accommodate spots that are difficult to segment. Flags are indicated by the software as red circles in default positions. The four categories of flagging are for low expression, failure to detect a spot, artifact noise, and irregular ellipse

shapes. The user decides whether to keep or dismiss categories of flagged spots.

Spots are flagged as having low expression if their intensity falls within a 95% confidence interval for the background estimate. To construct this interval, the program uses the background estimate specified in Section 2.3.4, and an empirical standard error based on the spread of the left tail of the background distribution, σ^-. A default circle is drawn and the average intensity of the pixels within is compared to the confidence interval. That is,

$$\text{flag} = 1 \text{ if } \sum_{(i,j)\in S^*} [r(i,j) + g(i,j)]/n_{S^*} < b^r + b^g + 1.96 * \sigma^-_{r+g}$$

where S^* is a default circle with radius r_A. If the average spot intensity falls outside this interval, its intensity is significantly different from background. This ensures a significant signal to background ratio for spot detection. Although spots with low expression are flagged, information can still be extracted from these regions. A default circle position is used to essentially measure intensity values of zero.

Occasionally, the spot detection algorithm fails to detect any signal at all. That is, either no signal exists or it is so small that it is likely to constitute artifact noise such as dust. The second flag category accounts for this. The current flag setting for this is a minimum detected ellipse height and width of three pixels.

Artifact noise on an array will often overlap a signal region. The difference between an artifact and real signal is difficult to detect objectively. Of course, it can also be difficult for human judgment to differentiate subjectively. Nevertheless, there are objective clues that can indicate the presence of artifact noise. The software application uses these clues for the third category of flags. Specifically, an algorithm looks for concavities in the cross-sectionals to flag spots for the presence of artifact noise.

Using the cross-sectionals, smoothed $x(i)$ and $y(j)$, the algorithm calculates second derivatives numerically, $x''(i) = x(i+2) - x(i+1) - [x(i+1) - x(i)]$. The sign is used to determine concavity in smoothed $x(i)$. If $x''(i) > 0$ for at least three pixels $i, i+1, i+2$, then the spot is flagged. The method looks for concavity as an indicator of artifact noise

because it implies the presence of multiple signals. Spots are also flagged when the pixel intensities within a spot have a "donut" shape as this also generates positive second derivatives.

The last category of automated flagging is an irregular ellipse. If the shape of the ellipse deviates too far from a circle, it is likely to have been influenced by noise factors and hence is flagged. Current protocol is that spots are flagged if their spot width is twice their spot height or vice versa. The four types of flags are illustrated in Figure 2.10.

2.3.6 Signal extraction and channel integration

The methods assume that images have been either simultaneously scanned or adjusted after sequential scanning so that channels are aligned. This assumption is used to add the two channels together so that greater information is used for grid alignment and spot detection. That is, let $z(i,j) = r(i,j) + g(i,j)$ and use these intensity values for the methods described in the above sections. Because cDNA clones are in the same location for hybridization to both mRNA samples, it follows that spot detection should find the regions of clone placement rather than regions of hybridized signal. Adding channels together during image analysis approximates detection of clone placement. Other methods for integrating channels discussed in Section 2.1.5, where spot detection is performed separately for each channel, do not use this logic.

After each spot region is defined, intensity values for each pixel within the ellipse are used to calculate summary measures. Currently, the summery measures exported are the mean and median signal values. These summary measures are then used to construct the standard ratio of the two channels. The standard deviation of the mean value is also exported. For now, the standard deviation estimate is the classic estimate assuming independent, identically distributed pixels values $\hat{\sigma}^2 = \sum_{(i,j)\in S}[z(i,j) - \bar{z}]^2/(n_S - 1)$. Standard error estimates accounting for spatial correlation between pixels are discussed in Chapters 3-5.

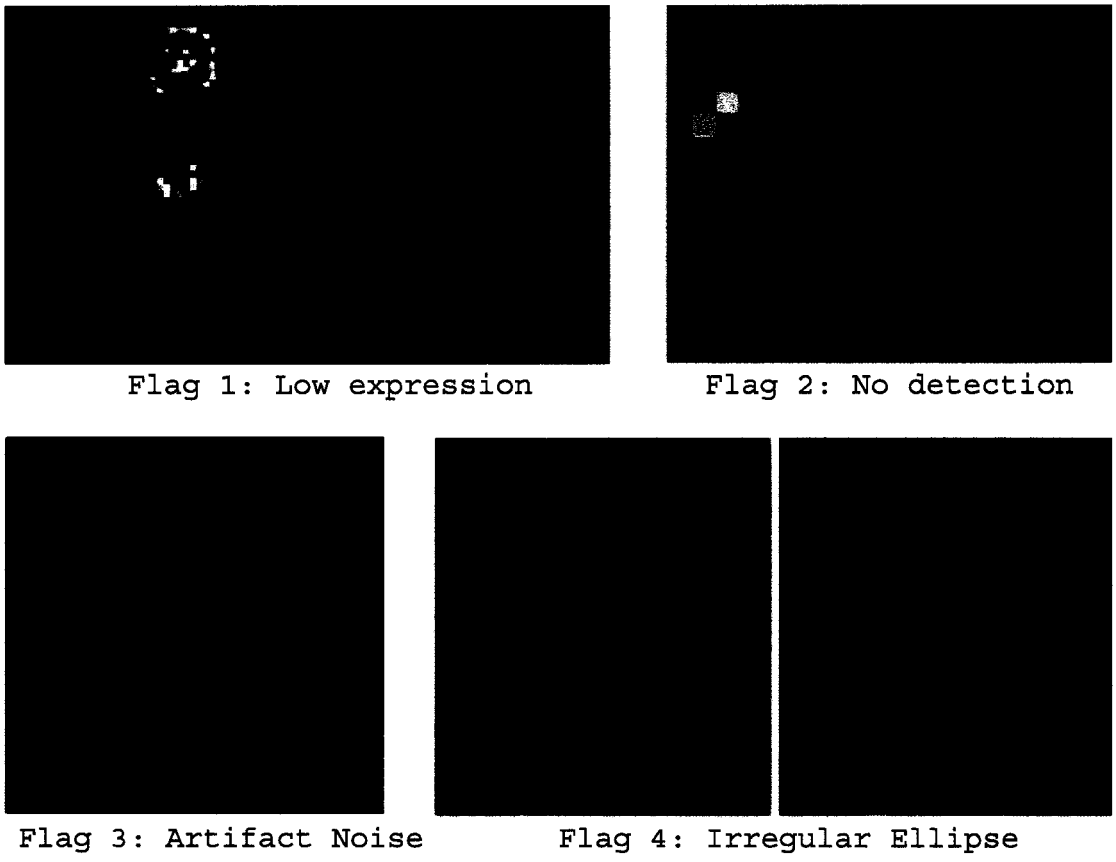


Figure 2.10: Examples of spots flagged by SignalViewer. Flag 1 identifies spots with low intensity values indicated here by red crosses. Other spots shown have sufficient intensity to allow for reliable spot quantification. Flag 2 shows a spot with nearby artifact noise. Spot detection falsely picks up this small bit of dust instead of the nearby spot. Because dust is usually only a few pixels in area, the ellipse is not large enough to be deemed reliable spot detection. Flag 3 displays a spot with a vertical scratch. The pixels with a lack of intensity through the center of the spot indicate the scratch. This type of artifact noise is easily identified because the projection is concave on the x-axis. Flag 4 shows a feature with a nearby spot overlapping the grid frame. As can be seen, the spot detection algorithm attempts to pick up both of these signals resulting in an irregular ellipse shape that is flagged.

2.4 Examples

To illustrate the methods, consider two examples of microarray image analysis in SignalViewer. Neither of these images were used in the development of the methods given in the previous section. The first will consist of a high quality image and the second will be of lesser quality. Both experiments were generated at the Fred Hutchinson Cancer Research Center core microarray facility. The experiments, with parameters given in section 2.1., were part of a dilution series set containing 96 clones from the yeast genome. Each of the 96 clones are replicated four times within a block and each block is replicated 16 times on the array. This means that for each experiment, there are 64 replicates of each gene clone resulting in 6,144 total spots.

Example 1, shown in Figure 2.1, compared a yeast strain at 100% concentration to itself at 50% concentration. Example 1 has only small levels of artifact noise. These artifacts show up as bright intensity streaks crossing several blocks of data and background. Image analysis was performed on Example 1 with SignalViewer and GenePix. The overall signal to background ratio (SBR) was 26.8, where this ratio is defined in Section 2.1. Slight adjustments to the automated grid alignment needed to be performed on five of 16 blocks. Three of these blocks were in the last row. The grids required only slight adjustments vertically by one or two pixels. No manual adjustments to the SignalViewer spot detection or flagging were performed. Comparisons with the GenePix signal extraction were made. Median values extracted from both packages resulted in a correlation coefficient of 0.86 within each channel. Sample spot detection is shown in Figure 2.11.

Example 2, shown in Figure 2.2, compared the yeast strain at 50% concentration to itself at 100% concentration. Examples 1 and 2 are dye swaps. The SBR for Example 2 is 14.73. While there was not significant artifact noise in this image, the overall expression levels are lower than in Example 1. No manual adjustment of grid blocks was required. No adjustments to spot detection or flagging were performed. Comparisons of median values yielded a correlation coefficient between SignalViewer and GenePix of 0.87 in the green

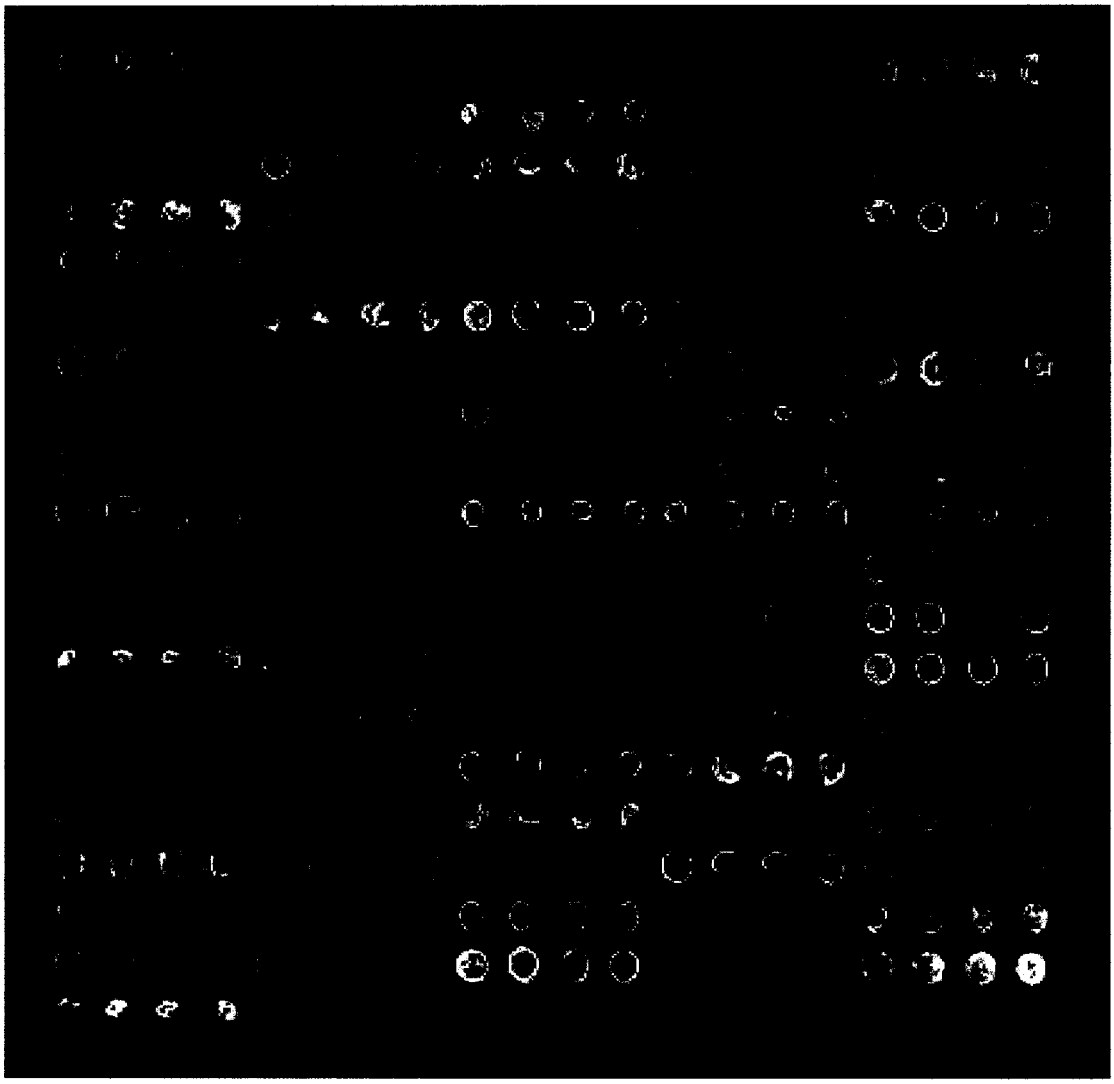


Figure 2.11: Spot detection for a sample block from Figure 1. Note that flagging routines successfully identify those spots with large artifacts, seen as bright red marks, overlapping the spots. Spot detection for all spots characterizes the region of interest well. Spots of low intensity are aptly flagged.

channel and 0.90 in the red channel. Sample spot detection is shown in Figure 2.12.

Figures 2.13 and 2.14 compare unadjusted ratios extracted from Example 2. Based on the figure and correlation estimates, there is substantial agreement between the two packages. Those spots with differing ratios between the image analysis packages can be accounted for with two variables: spot size and SignalViewer flagging. Two regions in these figures are of note. First, there is a cluster of spots deviating from the main trend with larger SignalViewer ratios than GenePix. These spots are unflagged and have spot sizes of less than 40 pixels. Because GenePix does not allow for spot sizes less than 52 pixels, the segmentation necessarily includes background pixels that can influence the resulting ratio. Second, spots that have large ratios differ substantially from GenePix to SignalViewer. Note that these spots are flagged by SignalViewer, for artifact noise specifically, and have default spot sizes of 137 pixels. In usual instances, these flagged spots could then be manually adjusted to capture appropriate signal regions.

Because Example 2 is a dilution experiment, the ratios are expected to center about $\frac{1}{2}$ instead of 1. Clearly the ratios are not $\frac{1}{2}$ for either image analysis package. The data requires normalization before matching expected ratio values for the dilution series.

Comparisons with GenePix show significant differences in the spots flagged. Table 1 shows the differences in flagging for Example 1 and Table 2 shows the differences for Example 2. In both cases, SignalViewer flags a larger number of spots. The following examines the distribution of flagging categories for spots flagged by SignalViewer. For Example 1, 893 spots were flagged. Of those also flagged by GenePix, 72% were flagged for low expression and 4% were flagged for artifact noise. Of those not flagged by GenePix, 14% were flagged for low expression and 64% were flagged for artifact noise. For Example 2, 969 spots were flagged. Of those also flagged by GenePix, 93% were flagged for low expression and 3% were flagged for artifact noise. Of those not flagged by GenePix, 54% were flagged for low expression and 44% were flagged for artifact noise. In both cases, there is a large number of spots with artifact noise flags in SignalViewer that are not flagged by GenePix.

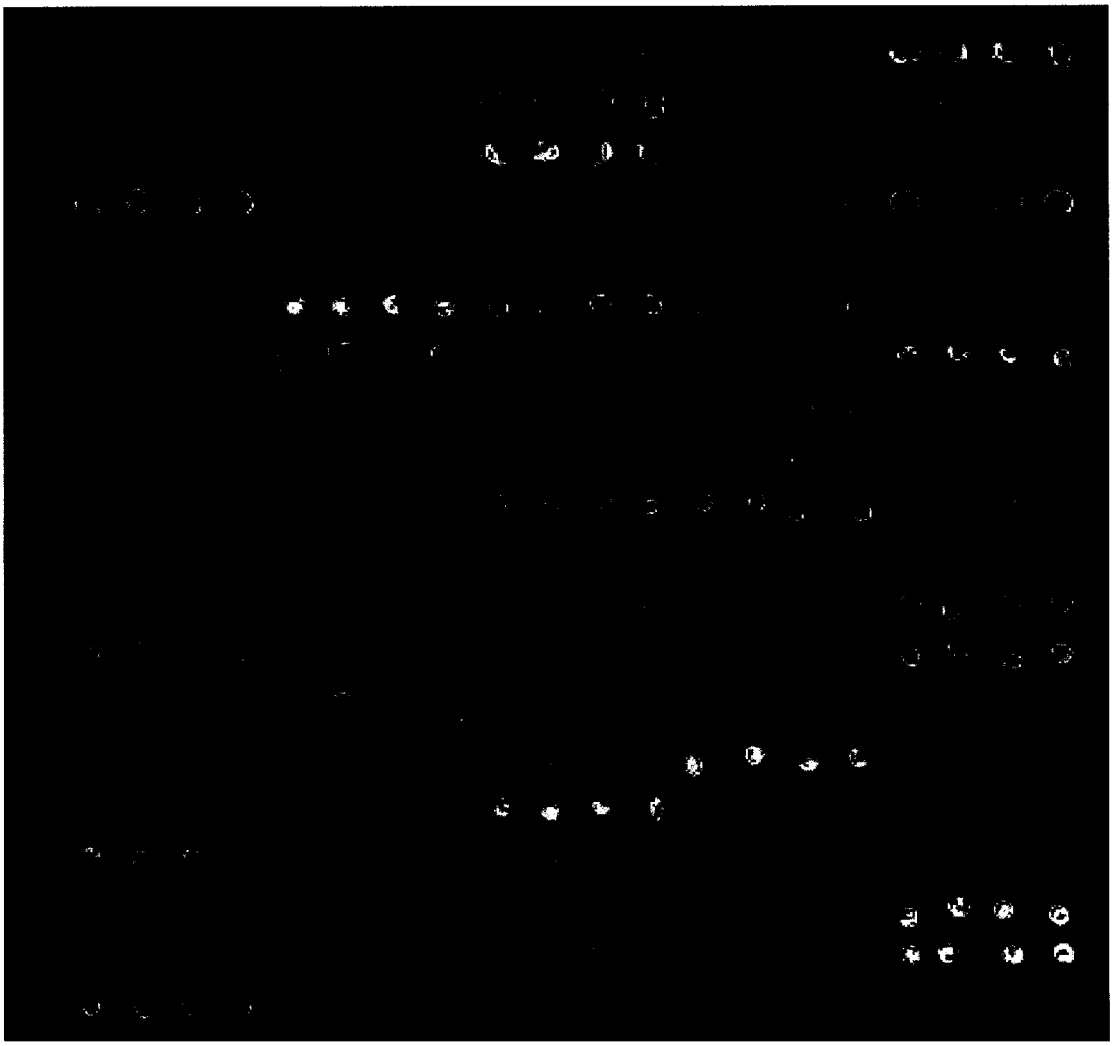


Figure 2.12: Spot detection for a sample block in Figure 2. Again, flags successfully identify spots of low expression. Spots with nearby artifact noise are also flagged.

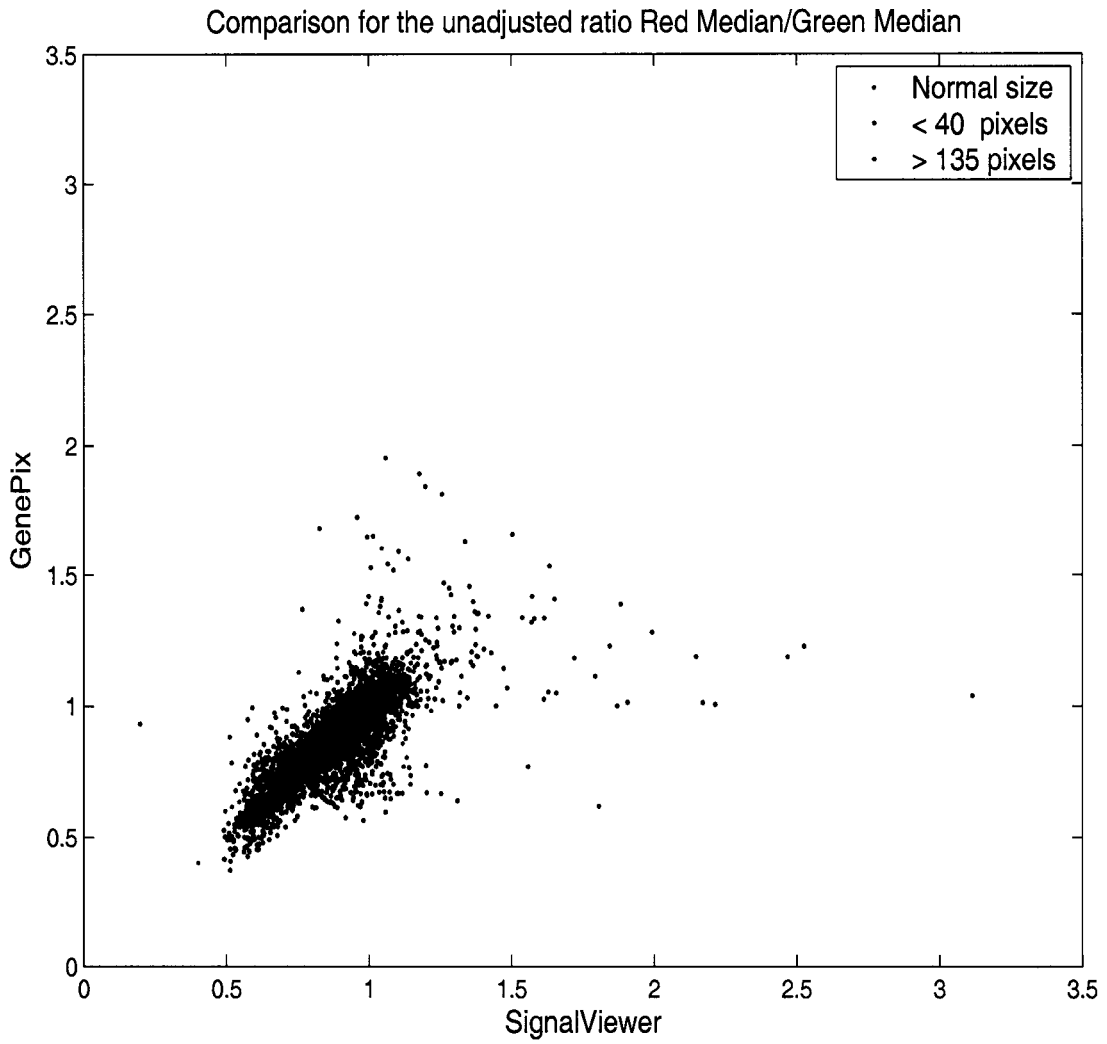


Figure 2.13: Unadjusted ratios calculated for Example 2 with colors indicating spot size categories.

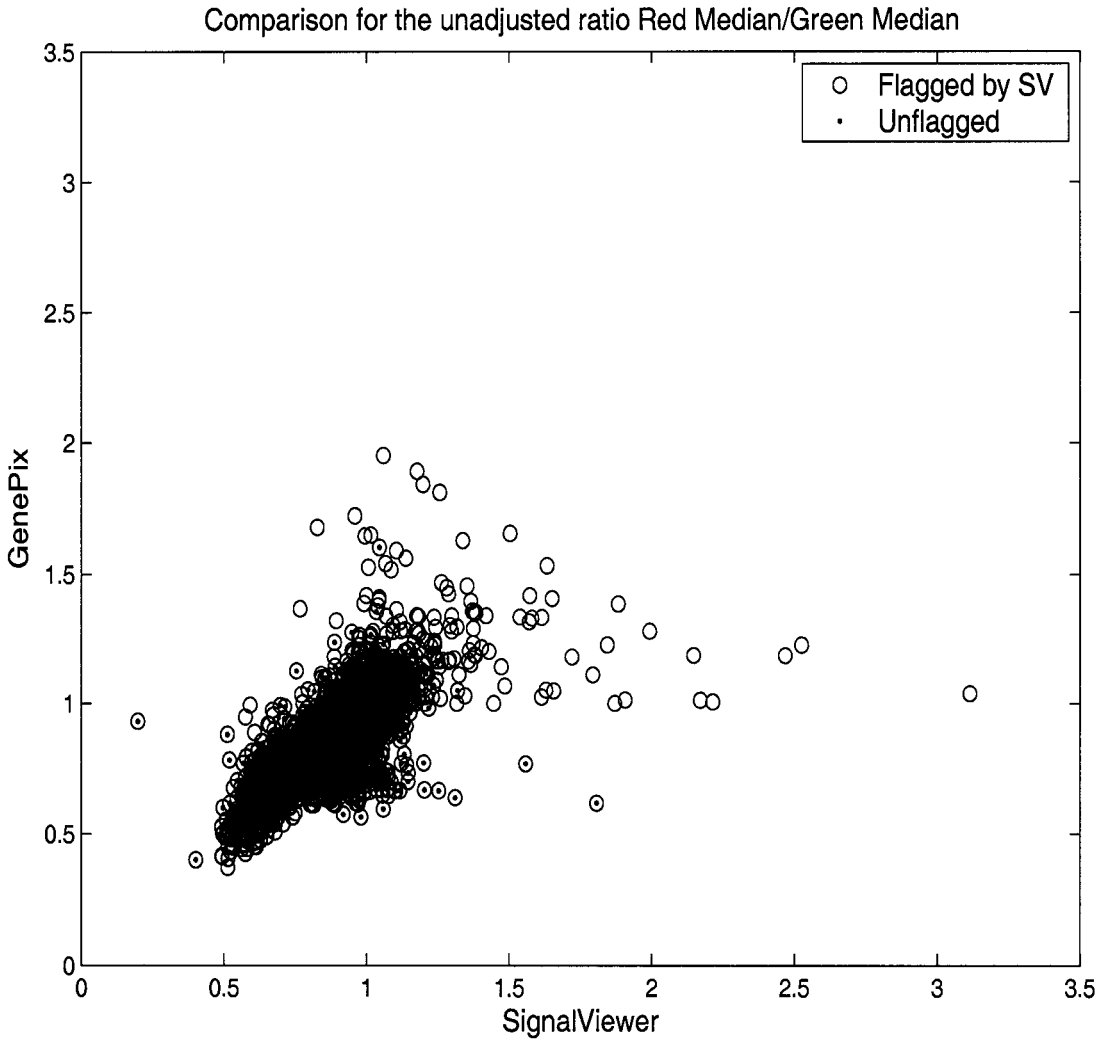


Figure 2.14: Unadjusted ratios calculated for Example 2 with circles indicating SignalViewer flagging.

Table 2.1: Flagging Procedures Comparison for Figure 2.1, SV stands for SignalViewer and GP for GenePix

	Not flagged by GP	Flagged by GP	
Not flagged by SV	5144	137	5251
Flagged by SV	847	46	893
	5961	183	6144

Table 2.2: Flagging Procedures Comparison for Figure 2.2, SV stands for SignalViewer and GP for GenePix

	Not flagged by GP	Flagged by GP	
Not flagged by SV	4982	193	5175
Flagged by SV	851	118	969
	5833	311	6144

Since spots are replicated four times within each block, the differences in unadjusted spot ratios can be compared to their replicate mean and will call it $\bar{D}$. Here, a difference is defined as the absolute value of each ratio of spot medians minus its replicate mean. This difference is then averaged to give a comparison of spot ratio estimates to their "true value". In Example 1, spots flagged by SignalViewer for artifact noise yield $\bar{D} = 0.0689$ while spots flagged by GenePix yield $\bar{D} = 0.0669$. The corresponding values in Example 2 are 0.0814 and 0.0577. In both examples, $\bar{D}$ is larger for the set of spots flagged for artifact noise by SignalViewer. Results for $\bar{D}$ are similar regardless of whether GenePix or SignalViewer is used to calculate the ratio estimate. In Example 1, for spots not flagged by SignalViewer, $\bar{D} = 0.0389$, while for spots not flagged by GenePix, $\bar{D} = 0.0415$. The corresponding values in Example 2 are 0.0354 and 0.0375. In both examples, $\bar{D}$ is smaller when the average is taken over the set of spots not flagged by SignalViewer. Thus, the conclusion is that SignalViewer was more successful at flagging incongruent spots.

Because the sample images replicate a gene 64 times over the array, a variance estimate of ratios for each gene can be calculated. These variance estimates can then be compared

between image analysis packages. Figure 2.15 shows this comparison for both sample images. For Example 1, the SignalViewer variance estimates are larger than GenePix on average. For Example 2, the converse is true. Cautiously drawing a conclusion from this, it seems that SignalViewer will perform better on the signals of lower intensity that dominate Example 2. But GenePix performed better on the more reliable data presented in Example 1.

Automated methods in SignalViewer are suitable when the SBR is one or greater. This threshold was determined by simulating simple images and testing the automated procedures. Background and signal values were simulated for images containing nine blocks with twenty rows and columns. The background data was sampled from an exponential distribution with an expected value of 100, $E(100)$. Signals were sampled from $E(SBR*100)$ with ever decreasing values for the SBR. Even when $SBR = 1$, SignalViewer was still able to perform automated grid alignment and spot detection, despite the fact that signal and noise levels were equal. This is because the methods in SignalViewer use the structure of a microarray image to assist in their analysis. Note that grids were not perfectly aligned for the image in Example 1 despite an SBR of 26. Arrays may have the same SBR value, but this does not imply that they will both be segmented successfully. This is because spatially determined artifacts of large size can have more of an impact on image analysis than the SBR.

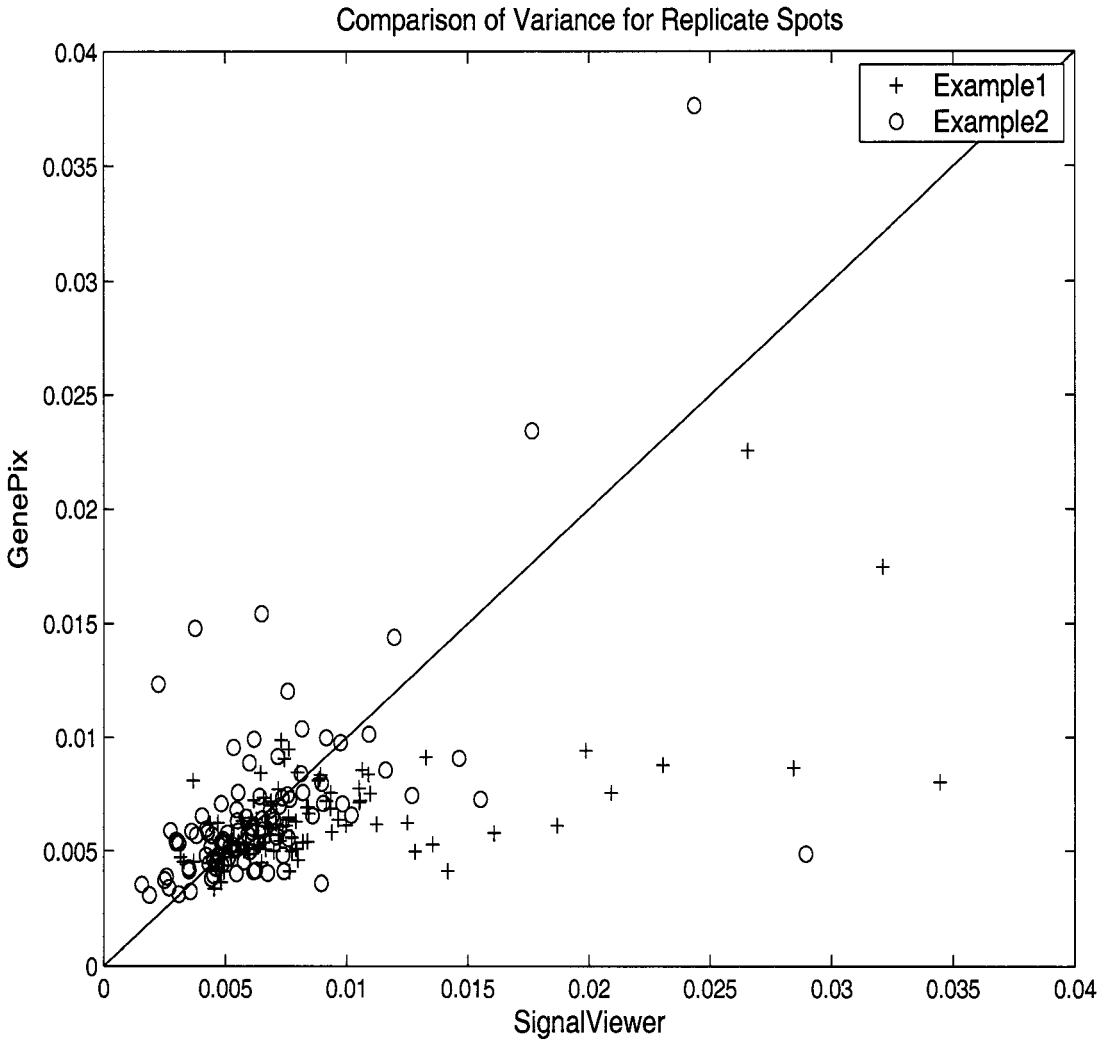


Figure 2.15: Gene variance estimates for Examples 1 and 2.

2.5 Discussion and Conclusions

SignalViewer contributes to the pool of existing microarray image analysis techniques by developing algorithms that are statistically driven and hence, objective and reproducible. This chapter provides well-defined methods for grid alignment, spot detection, background estimation, spot flagging, and signal extraction. These methods have been integrated into a software application. A compiled version for Windows can be downloaded as an executable file from http://qge.fhcrc.org/signalviewer.

Methods introduced that are unique to this paper include the following four contributions.

- Explicit automated grid alignment procedures.
- Ellipses of any width or height within the frame used to define spot regions.
- Meaningful and intuitive channel integration for image analysis.
- Automated flagging classified into four informative categories. This provides flexibility for downstream analysis. For example, a researcher can choose to examine data from spots with low expression values but not from spots with artifact noise. Because SignalViewer exports the data with five different values for the flag variable, users can eliminate only those genes that have been flagged for artifact noise while keeping data flagged for other reasons.

Background estimation for SignalViewer is similar to the methods reported by the NIH microarray group for their image analysis software QuantArray. That is, a mode estimate is used and variance of this estimate is determined by the spread of the empirical distribution. The method of grid alignment resembles the peak identification of Brändle et al [5]. The percentile projections that are used, however, are more robust than the Radon transform projection cited. The spot detection methods strike a balance between structured and

unstructured segmentation. GenePix imposes a circular structure on array spots with one of five possible sizes. SignalViewer allows for elliptical shapes of any size. Programs such as Spot and Dapple impose no structure on array spots therefore allowing for shapes that are clearly not realistic, particularly for spots with low expression. SignalViewer imposes some structure to ensure the integrity of spot segmentation when dealing with low intensities. Further, manual overrides are in place for grid alignment and spot detection in the event of algorithm errors or ambiguous data. Signal extraction and channel integration methods are similar to GenePix. But GenePix uses a ratio image for spot detection and channel integration, while SignalViewer performs spot detection on the addition of the red and green channels. Like most other programs, SignalViewer outputs mean and median intensity values for each spot within each channel.

The previous section gave correlation estimates for similarities in SignalViewer and GenePix values. If the methods were exactly the same, a correlation of nearly one is expected. But, a correlation of 0.87 indicates that there is substantial deviation between the two different software packages. The correlation score, as an average, does not indicate the number of spots where intensity values are variable between the two packages. Figure 2.13 gives a better idea of the similarity in signal intensity for the two software applications. Outlying points need to be resolved. The following examines further the the impact that differences in segmentation have on signal intensity values. Figure 2.16 provides five different segmentations of the same sample spot. The first segmentation is that performed by SignalViewer. The four remaining segmentations represent possible segmentations that a lab user might specify. The five image analyses result in green channel mean signal intensities fluctuating from 3083.6 to 3714.3 and red channel mean intensities fluctuating from 4466.3 to 5616.3. Green median intensities range from 2998 to 3606 and red median intensities range from 4081 to 5818. For all values, SignalViewer segmentation resulted in the largest values, indicating that the application detects the brightest region. Despite fluctuations in mean intensity values, the ratio of means was relatively constant at 1.5 across the five

examples. The ratio of medians, however, was more variable, ranging from 1.311 to 1.642. An investigation into the type of estimate used to measure spot intensity would be valuable. That is, it is important to determine whether ratio estimates should be consistent despite differences in image analysis or deviate to correspond with these differences.

Further improvements that can be made to these methods include a scheme to rotate grid alignment when necessary. Often these rotations are on the order of two to three degrees and take place for the entire array. Automated channel alignment for sequential scanners could also be incorporated into the program. In future work, estimates that describe the relative mRNA expression for two tissues will also be explored. The attempt will be to develop a more stable and well-defined estimate than the log ratio currently used.

In Chapter 3 through 5, means to describe data quality for cDNA microarrays are explored. Currently, the standard deviation estimate is the classic estimate assuming independent, identically distributed pixels values $\sum_{(i,j)\in S}[z(i,j) - \bar{z}]^2/(n_S - 1)$. It is certainly the case that pixels within a spot are not independent entities. Chapter 3 will investigate semi-parametric methods to describe correlation between pixels. Chapters 4 and 5 will explore parametric and non-parametric methods of spot quality on both real and simulated data. Chapter will conclude with a discussion of the ways in which reliability estimates can be used in downstream analysis.

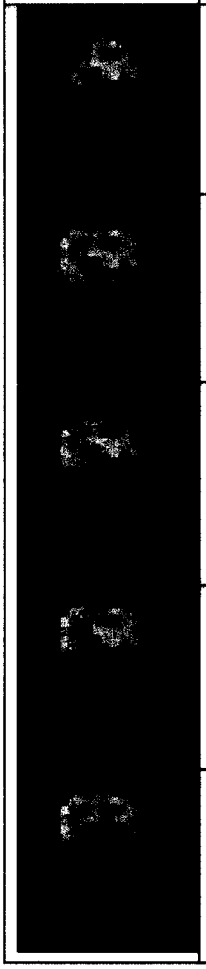
	Green			Red				
Segmentation	Mean	Median	Standard Error	Mean	Median	Standard Error	R/G	med(R)/ med(G)
	3714.3	3606	1726.76	5616.3	5818	2596.28	1.512	1.613
	3252.8	3349	2136.74	4923.2	5500	3203.22	1.514	1.642
	3083.6	2998	1823.47	4466.3	4129	2847.68	1.448	1.377
	3488.2	3315	1662.62	5295.9	5277	2588	1.518	1.592
	3473.1	3113	1398.41	4952.7	4081	2123.17	1.426	1.311

Figure 2.16: The above table shows five examples of possible spot segmentation for one given feature. Means, medians, and standard errors output by SignalViewer are provided for each channel and segmentation. The last two columns provide the ratio of mean values and ratio of median values without adjustment for background.

Chapter 3

ESTIMATING EQUATIONS TO DESCRIBE WITHIN-SPOT VARIANCE

3.1 Background

Because microarray technology is an error prone data collection process, it is of interest to quantity the reliability of the platform. It has been shown previously that measures of spot quality are related to the consistency and distribution of log-ratio outcomes in [50] and [51]. In these two papers, a signal quality measure called q_{com} is introduced and analyzed. The measure incorporates five different attributes from the data: size, signal-to-noise ratio, background uniformity, high background intensity, and pixel saturation.

- Size irregularity is measured by

$$q_{\text{size}} = \exp\left(-\frac{|A - A_0|}{A_0}\right)$$

  where A is the area of the spot and A_0 is the average spot area on the array. Wang et al argue that smaller spots should be penalized as they likely indicate isolated noise. Larger spot size might mean that contaminants or other spots lie in close proximity.

- Signal-to-noise ratios are measured by

$$q_{\text{sig-noise}} = \bar{Y}/(\bar{Y} + b_Y)$$

  where $\bar{Y}$ is the mean spot signal and b_Y is a local background estimate.

- Two background attributes are used, q_{bkg1} and q_{bkg2}. If a_1 is a normalizing constant

such that $\max(q_{\text{bkg1}}) = 1$, then $q_{\text{bkg1}} = a_1/\text{CV}_{\text{bkg}}$ and CV_{bkg} is the local background coefficient of variation. Let a_2 similarly be a normalizing constant for q_{bkg2}, and $q_{\text{bkg2}} = a_2\{\text{bkg}_0/(\text{bkg}_0 + \text{bkg}_1)\}$ where bkg_0 is the global average of background estimates and bkg_1 is the local background estimate.

- Finally, a measure of spot saturation is used: $q_{\text{sat}} = I(\text{if saturated pixels} < 10\%)$.

A geometric mean is used to combine the five quality scores: $q_{\text{com}} = (q_{\text{size}} \times q_{\text{sig-noise}} \times q_{\text{bkg1}} \times q_{\text{bkg2}})^{1/4} \times q_{\text{sat}}$. And another geometric mean is used to combine quality information from both channels: $q_{\text{com}} = \sqrt{q_{\text{com,R}} \times q_{\text{com,G}}}$. The larger the value of q_{com}, the better the spot quality. This final spot score is shown to have a relationship with the variability of log-ratios and the correlation and consistency between spot replicates.

The largest problem with the q_{com} score is its lack of objectivity. The score highlights five issues and the choices to assess each issue are somewhat arbitrary and unjustified. Each microarray expert will have their own opinion about what variables describe a bad spot. For example, the Spot package developed by Yang et al uses a circularity statistic to describe deviation of a segmented spot from a perfect circle [54]. Researchers at Applied Precision Inc. suggest using an inverse coefficient of variation to describe signal reliability [6]. This statistic is different than q_{bkg1} described above as it examines the CV of the ratio estimate instead of the background. Specifically, let i denote a spot, and $Z_i = (\bar{R}_i - b_i^R)/(\bar{G}_i - b_i^G)$ be the spot ratio. Assuming normality, a variance estimate is derived for Z_i,

$$\sigma^2_{Z_i} = \sigma^2_{G_i}\frac{\bar{R}_i^2}{\bar{G}_i^4} + \frac{\sigma^2_{R_i}}{\bar{G}_i^2} - 2\sigma_{RG_i}\frac{\bar{R}_i}{\bar{G}_i^3}$$

so that $CV = Z_i/\sigma_{Z_i}$. The inverse of this spot quality score is used in the paper to estimate a signal to noise ratio. This score assumes that spots follow a normal distribution and that pixels are independent, neither of which seem feasible for array data.

Finally, a discussion of signal quality measures exists in recommendations by Handran and Zhai of Axon Instruments, Inc [25]. In an application note, they describe several

different variables and ad hoc ways in which they can be used to assess data reliability. Their first suggestion is to compare (1) the ratio of median estimates, (2) the median of pixel level ratios and (3) the ratio based on a regression line. They claim that a discrepancy between these three measures indicates a spot of bad quality. But, they do not show any evidence for why this is known to be true. It can be argued that a spot is reliably measured with one of the three ratios, but not the other two, and therefore should not be penalized for bad quality. Similarly, looking at the deviance between the background means and background medians is recommended. Other suggestions are to examine the spot standard deviations within each channel, the R^2 measure from the regression ratio, the percent of saturated pixels within each channel, or the percent of foreground pixels greater than one or two standard deviations above the background. Finally, Axon observes that spots where the sum of red and green estimates are small have less reliable ratio estimates. This has been noted in many studies showing a "fish shape" in M versus A plots [54]. All of these suggestions to check for spot quality are simply guidelines and not hard and fast rules to define faulty data. The report does not provide any evidence for relationships between output statistics and true reliability.

None of the above-mentioned strategies use previous assessments of quality classifications to develop their spot quality metrics. Rather, these approaches are *ab initio* or based on intuition and reasoning. There is, of course, potential for classification schemes based on the experienced assessment of lab researchers that will train an algorithm for quality prediction. For example, a group of microarray researchers could be shown a wide variety of spot images and asked to score each one for quality. These scores could then be used to train an algorithm such as a SVM (support vector machine) or a CART model (unpublished preliminary research by Jinbo Chen and Steve Self). The disadvantages of these predictive models are (1) the need for many man hours which lab scientists are reluctant to contribute, (2) uninterpretibility of "black box" approaches like SVMs, and (3) the potential bias induced by the types of array data used or the lab researchers assessing the data.

Based on the above review, the author concludes that a more objective approach is to use an *ab initio* spot quality metric that investigates the variance within each spot on an image, while making the fewest possible assumptions. In the remaining chapters of this dissertation, consistent and objective means of describing spot quality via within-spot variance estimates will be developed. This query will begin with a semi-parametric model applying estimating equations.

3.1.1 Estimating Equations

Theory for estimating equations published by Liang and Zeger allows for the analysis of correlated data in a regression setting [34]. This theory is most often applied to longitudinal data studies where several measurements on the same individual are analyzed. Each individual then contains a cluster of measurements that are correlated, but the individuals are independent. Suppose that n is the number of individuals in a study. Estimating moments from generalized linear models involves solving the score equation

$$U(\beta) = \sum_{i=1}^{n} \frac{\partial \mu_i}{\partial \beta} \mathbf{V}_i^{-1} \{Y_i - \mu_i(\beta)\} = 0.$$

The covariance for the consistent solution to the generalized linear model, $\hat{\beta}$, is

$$\begin{aligned} \mathrm{Cov}(\hat{\beta}) &= \left(\tfrac{1}{n} E\left[\textstyle\sum_{i=1}^{n} \frac{\partial U_i}{\partial \beta}\right]\right)^{-1} \tfrac{1}{n} \mathrm{Var}\left\{\textstyle\sum_{i=1}^{n} U_i(\beta)\right\} \left(\tfrac{1}{n} E\left[\textstyle\sum_{i=1}^{n} \frac{\partial U_i}{\partial \beta}\right]\right)^{-1} \\ &= \left(\left(\frac{\partial \mu}{\partial \beta'}\right)' \mathbf{V}^{-1} \left(\frac{\partial \mu}{\partial \beta'}\right)\right)^{-1} \left(\frac{\partial \mu}{\partial \beta'}\right)' \mathbf{V}^{-1} \mathrm{Cov}(Y) \mathbf{V}^{-1} \left(\frac{\partial \mu}{\partial \beta'}\right) \left(\left(\frac{\partial \mu}{\partial \beta'}\right)' \mathbf{V}^{-1} \left(\frac{\partial \mu}{\partial \beta'}\right)\right)^{-1} \\ &= I_n^{-1}(\beta) J_n(\beta) I_n^{-1}(\beta). \end{aligned}$$

This covariance is estimated by

$$\hat{\mathrm{Cov}}(\hat{\beta}) = \left(\left(\frac{\partial \mu}{\partial \beta'}\right)' \hat{\mathbf{V}}^{-1} \left(\frac{\partial \mu}{\partial \beta'}\right)\right)^{-1} \sum_{i=1}^{n} U_i(\hat{\beta}) U_i(\hat{\beta})' \left(\left(\frac{\partial \mu}{\partial \beta'}\right)' \hat{\mathbf{V}}^{-1} \left(\frac{\partial \mu}{\partial \beta'}\right)\right)^{-1}$$

and is consistent even when $\mathbf{V}$ is misspecified. Proper specification of $\mathbf{V}$ yields greater efficiency in the variance estimates.

3.1.2 Estimating Equations Applied to Microarray Spots

Chapter 2 illustrates a method for spot segmentation for microarray data. Here, models to quantify spot signals will condition on the segmentation results. A simple working model to describe spot signals is

$$E(Y_{ij}|S) = \beta_0 + \beta_1 I((i,j) \in S) + \epsilon_{ij} \tag{3.1}$$

where i and j indicate pixel location, S is the fixed spot location, and $\epsilon_{ij} \sim (0, \mathbf{V})$. Y_{ij} is an intensity measure at pixel (i,j) that can represent, for example, intensity in the red channel of the microarray, the green channel, the sum of the two channels, or the ratio of the two channels. Spatial correlation between pixels can be accounted for in the off-diagonal elements of the variance matrix $\mathbf{V}$.

Under the model described in (3.1), where $\mathbf{X}_{ij} = [1, I((i,j) \in S)]$, the estimating equations described in Section 1.1 simplify to

$$\begin{aligned} \hat{\vec{\beta}} &= (\mathbf{X}'\hat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}'\hat{\mathbf{V}}^{-1}Y \text{ and} \\ \hat{\text{Cov}}(\hat{\beta}) &= (\mathbf{X}'\hat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}'\hat{\mathbf{V}}^{-1}\vec{r}\cdot\vec{r}'\hat{\mathbf{V}}^{-1}\mathbf{X}(\mathbf{X}'\hat{\mathbf{V}}^{-1}\mathbf{X})^{-1} \end{aligned}$$

where $\vec{r} = \vec{Y} - \hat{\vec{\mu}}$. Infinite options are available to determine $\hat{\mathbf{V}}$. For this research, three formulations of $\mathbf{V}$ were considered. Suppose that V has off-diagonal elements $\gamma_{ij,kl}$ such that, for Euclidean distance $D_{ij,kl} = \sqrt{(i-k)^2 + (j-l)^2}$ and a fixed neighborhood size B,

$$\gamma_{ij,kl} = (\alpha_0 + \alpha_1 D_{ij,kl}) I(D_{ij,kl} \leq B) \tag{3.2}$$

$$\gamma_{ij,kl} = \exp(-\alpha_1 D_{ij,kl}) I(D_{ij,kl} \leq B) \tag{3.3}$$

$$\gamma_{ij,kl} = \alpha e^{-D_{ij,kl}}. \tag{3.4}$$

Table 3.1: Illustration of Euclidean distance $D_{ij,kl} = \sqrt{(i-k)^2 + (j-l)^2}$ (on the left) and absolute distance $D_{ij,kl} = \max\{|i-j|, |k-l|\}$ (on the right).

$\sqrt{13}$	$\sqrt{8}$	$\sqrt{5}$	2	$\sqrt{5}$	$\sqrt{8}$
$\sqrt{10}$	$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
3	2	1	Y	1	2
$\sqrt{10}$	$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
$\sqrt{13}$	$\sqrt{8}$	$\sqrt{5}$	2	$\sqrt{5}$	$\sqrt{8}$

3	2	2	2	2	2
3	2	1	1	1	2
3	2	1	Y	1	2
3	2	1	1	1	2
3	2	2	2	2	2

Each of these formulations for the spatial correlation comes with a unique caveat. Equation (3.2) will result in a positive definite $\mathbf{V}$ only for a fixed subset of (α_0, α_1) values. The larger the value of B, the more limiting the restrictions on (α_0, α_1). Equation (3.3) does not allow for negative correlation and equation (3.4) only allows for a fixed level of decay with Euclidean distance.

As a side note about distance metrics, there are several choices for measuring distance between pixel pairs. The standard is to use Euclidean distance as given above. But another alternative, that will be used in the next chapter, is defined by $D_{ij,kl} = \max\{|i-k|, |j-l|\}$. Tables 3.1 shows the differences in the way these two distances measure pixel pairs.

The algorithm to solve for $(\hat{\vec{\beta}}, \hat{\vec{\alpha}})$ in any one of the three spatial correlation formulations is

1. Initialize $\hat{\vec{\beta}}_{(0)} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'Y$.

2. Solve for $\vec{r} = y - \mathbf{X}\hat{\vec{\beta}}$.

3. Using the fact that $E(r_{ij} \cdot r_{kl}) = \gamma_{ij,kl}$, solve for $\hat{\alpha}$ using linear regression.

4. Form $\mathbf{V}_{\hat{\alpha}}$.

5. Update $\hat{\vec{\beta}}_{(1)} = (\mathbf{X}'\hat{\mathbf{V}}^{-1}\mathbf{X})^{-1}\mathbf{X}'\hat{\mathbf{V}}^{-1}Y$.

6. Continue to convergence $|\hat{\vec{\beta}}_{(i)} - \hat{\vec{\beta}}_{(i+1)}| \leq \epsilon$.

3.1.3 Limitations of Estimating Equations Applied to Spot Data

Theory developed in Liang and Zeger relies on replication over grouped observations, i.e., where $n \to \infty$ [34]. The microarray spot model proposed in Section 1.2 only has one cluster of correlated data and therefore, no replication over groups. The sandwich estimators for $\hat{\text{Cov}}(\hat{\beta})$ will fail when $n = 1$. Let i, j represent measurements within one replicate, in this case pixels within a spot. Note then that

$$\begin{aligned} \hat{J}_n(\hat{\beta}) = \hat{J}_1(\hat{\beta}) &= \tfrac{1}{m}\textstyle\sum_{i,j=1}^{m} U_i(\hat{\beta})U_j(\hat{\beta})' \\ &= \tfrac{1}{m}\left(\textstyle\sum_{i=1}^{m} U_i(\hat{\beta})\right)\left(\textstyle\sum_{i=1}^{m} U_i(\hat{\beta})\right)' \\ &\equiv \mathbf{0} \end{aligned}$$

because by definition of $\hat{\beta}$, $U(\hat{\beta}) = 0$.

The above result tends to be counterintuitive at first glance. Hence this result will be shown for the special case in (3.1), where $\mathbf{V} = \mathbf{I}$ is working independence. Recall that $\mathbf{X}_{ij} = [1, I((i,j) \in S)]$ and then

$$\begin{aligned} \hat{\text{Cov}}(\hat{\beta}) &= (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\vec{r}\cdot\vec{r}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1} \\ &= (\mathbf{X}'\mathbf{X})^{-1}\begin{pmatrix} \left(\sum_{(i,j)} r_{ij}\right)^2 & \sum_{(i,j)} r_{ij}\sum_{(i,j)\in S} r_{ij} \\ \sum_{(i,j)\in S} r_{ij}\sum_{(i,j)} r_{ij} & \left(\sum_{(i,j)\in S} r_{ij}\right)^2 \end{pmatrix}(\mathbf{X}'\mathbf{X})^{-1} \\ &= \tfrac{1}{n-n_S}\begin{pmatrix} 1 & -1 \\ -1 & \frac{n}{n_S} \end{pmatrix}\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}\tfrac{1}{n-n_S}\begin{pmatrix} 1 & -1 \\ -1 & \frac{n}{n_S} \end{pmatrix} \\ &\equiv \mathbf{0} \end{aligned}$$

because both $\sum_{(i,j)} r_{ij} = 0$ and $\sum_{(i,j)\in S} r_{ij} = 0$.

When the pixel data within an individual microarray spot is not replicated, sandwich estimators based on the model in (3.1) will collapse. Possible solutions to deal with the lack of spot replication would be to build sandwich estimators for microarrays with replicate spots printed. This would, however, limit any spot quality estimates that are derived to

array designs with spot replicates.

3.2 Weighted Estimating Equations Applied to Spot Data

The failure of the robust sandwich estimates for time series or image data is discussed in [33] and [35] among others. These authors recommend adjustments of one form or another to the estimate of $\text{Cov}(Y)$ in $\hat{\text{Cov}}(\hat{\beta})$. Generally speaking, the adjustments induce independence in the covariance matrix such that it will imitate replication. More specifically, this adjustment to $\hat{J}(\hat{\beta})$ as suggested in Lumley and Heagerty will look like

$$\hat{J}(\hat{\beta}) = \frac{1}{m}\sum_{i,j=1}^{m} w_{ij} U_i(\hat{\beta}) U_j(\hat{\beta})'$$

where $w_{ij} \to 1$ as $m \to \infty$ and $w_{ij} \to 0$ as $d_n(i,j) \to \infty$. This estimate is proven to be consistent [35].

Applying weighted estimating equations to the model in (3.1) simplifies notation quite a bit. If working independence is assumed for $\mathbf{V}$, then

$$\hat{J}(\hat{\beta}) = \mathbf{X}'\mathbf{w} \odot \vec{r} \otimes \vec{r}\mathbf{X}$$

where $\mathbf{w}$ is the matrix consisting of weights $w_{ij,kl}$, $\odot$ is the Hadamard product, and $\otimes$ is the Kronecker product. Note that $w_{ij,kl}$ is a weight on the covariance between two pixels in two-dimensional space. $\hat{J}(\hat{\beta})$ then looks like a matrix with the following components:

$$\begin{pmatrix} \sum_{(i,j)}\sum_{(k,l)} w_{ij,kl} r_{ij} r_{kl} & \sum_{(i,j)}\sum_{(k,l)\in S} w_{ij,kl} r_{ij} r_{kl} \\ \sum_{(i,j)\in S}\sum_{(k,l)} w_{ij,kl} r_{ij} r_{kl} & \sum_{(i,j)\in S}\sum_{(k,l)\in S} w_{ij,kl} r_{ij} r_{kl} \end{pmatrix}$$

3.2.1 *Classes of weight estimates*

The simplest of weights is an indicator dependent on distance, first proposed in [53]. In the case of spot data, $w_{ij,kl} = I\left(D_{ij,kl} < B\right)$. This means that the covariance between pixel

intensities drops to zero after a given Euclidean distance and is otherwise estimable by the product of residuals. This weight is referred to as the White-Domowitz throughout.

The most commonly known weight is the Newey-West estimator [41]. Newey and West noted that the previous estimate may not result in positive semidefinite variance matrices. As an alternative, they propose $w_{ij,kl} = 1 - \frac{D_{ij,kl}}{B+1}$. For both of these classes of weight estimates, B is based on prior knowledge about the correlation structure of the data.

The next section discusses the correlation structure for microarray spots. Nevertheless, a model where fewer assumptions are made about B is preferred. The third weight in this section, developed by Lumley and Heagerty, seems to fit this requirement [35]. These weights are called weighted empirical adaptive variance estimators (WEAVEs). Less reliant on the neighborhood bound, B, the WEAVE weight is an adaptive method that incorporates an estimate of the correlation structure, $\hat{\rho}$. The two suggested forms for the WEAVE estimate are $w_{ij,kl} = 1 \wedge Cn\rho_{ij,kl}^2$ and $w_{ij,kl} = I\{n\rho_{ij,kl}^2 > C\}$. Here, C is an ad hoc tuning constant. This estimate also accounts for the truncation bias that occurs when correlations are either omitted or down-weighted.

Centering bias occurs in the weighted variance estimator due to the evaluation at $\hat{\beta}$ instead of β_0. Lumley and Heagerty suggest correcting for this bias with an overall weight $w^* = 1/(1-\bar{w})$ [35]. This correction is implemented for all weight estimators used in Section 3.2.3 for simulation studies.

3.2.2 Correlation structure for spot data

By examining empirical correlation coefficient estimates from microarray experiments, it is possible to speculate about the type of weight that might be suitable for a robust model of the covariance. This is done by examining spots from a sample yeast array generated in Jeff Delrow's lab at the Fred Hutchinson Cancer Research Center. Within each spot, correlation estimates are calculated for a fixed Euclidean distance between pixels. Of interest is the distance at which correlation drops to zero. An ensuing series of plots show correlation

estimates within each spot for a given Euclidean distance. These figures apply to intensities from the red channel only. Results are similar for the green channel and combinations of both channels (not shown).

Figures 3.1 and 3.2 show the relationship between correlation and distance for spot and background pixels respectively. In both regions, correlation decays to zero at approximately eight pixels. The boxplots in Figures 3.3 and 3.4, however, indicate much variation within each distance for the correlation estimates across spots. This variation includes many instances of negative correlation as well. Note also that correlation levels increase again after twelve pixels in distance. This may be the true nature of the data, but it is more likely that the paucity of pixel pairs for large distance results in wilder correlation estimates.

3.2.3 Simulation study of weighted estimating equations

Spots were simulated from a multivariate normal distribution $\vec{Y} \sim MVN(\vec{\mu}, \sigma^2\mathbf{V})$, where $\mathbf{V}$ has components $\gamma_{ij,kl} = \exp(-\alpha_0 - \alpha_1 D_{ij,kl})$. The vector $\vec{\mu}$ imitates spot data with a pulse model having components $\mu_b + \mu_s \cdot I((i,j) \in S)$. μ_b represents background pixels and μ_s represents signal pixels. S is a circle with a radius of five pixels and the entire data frame is square of 20 by 20 pixels. So, $\vec{Y}$ has 400 pixel intensities.

For the purposes of this simulation study, let $\mu_b = 1000$, $\mu_s = 5000$, and $\sigma^2 = 3000$. Then, the studies investigate the performance of the three weight estimates discussed in Section 3.2.1 for various levels of (α_0, α_1). The first case will be $(\alpha_0, \alpha_1) = (0.8, 1.0)$ corresponding to weak correlation decaying to zero at about five pixels. The second case is $(\alpha_0, \alpha_1)(0.3, 1.0)$ and most closely resembles the structure seen in actual microarray data. The third case is $(\alpha_0, \alpha_1) = (1.5, 0.0)$, where the correlation is constant over distance with a value of roughly 0.22.

Tables 3.2, 3.3, and 3.4 display the results of three simulations for the White-Domowitz estimate. For each case, the number of simulations was $N = 200$. For all three simulations, the estimates for $\hat{\beta}$ are good as expected. In Table 3.2 when $(\alpha_0, \alpha_1) = (0.8, 1.0)$, the average

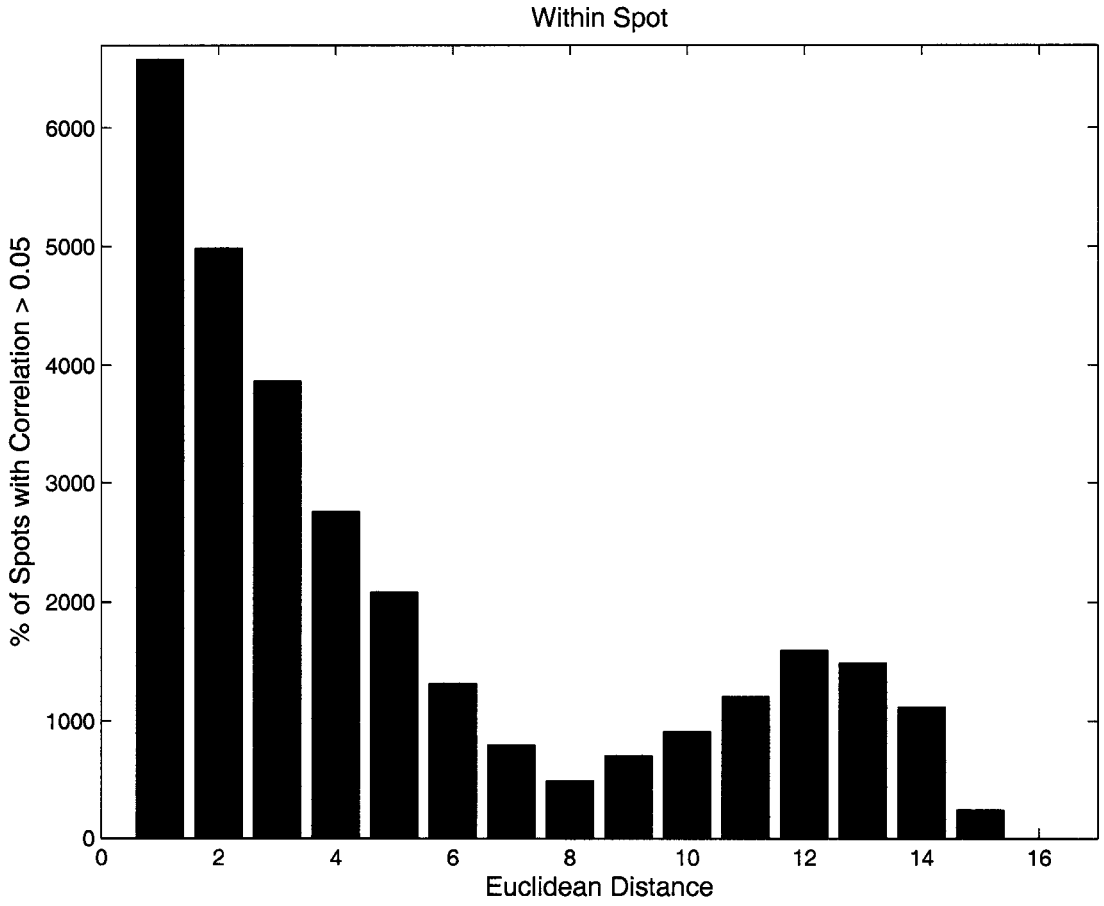


Figure 3.1: For pixels pairs at a given Euclidean distance *inside* spot regions, the number of spots with correlation above 0.05.

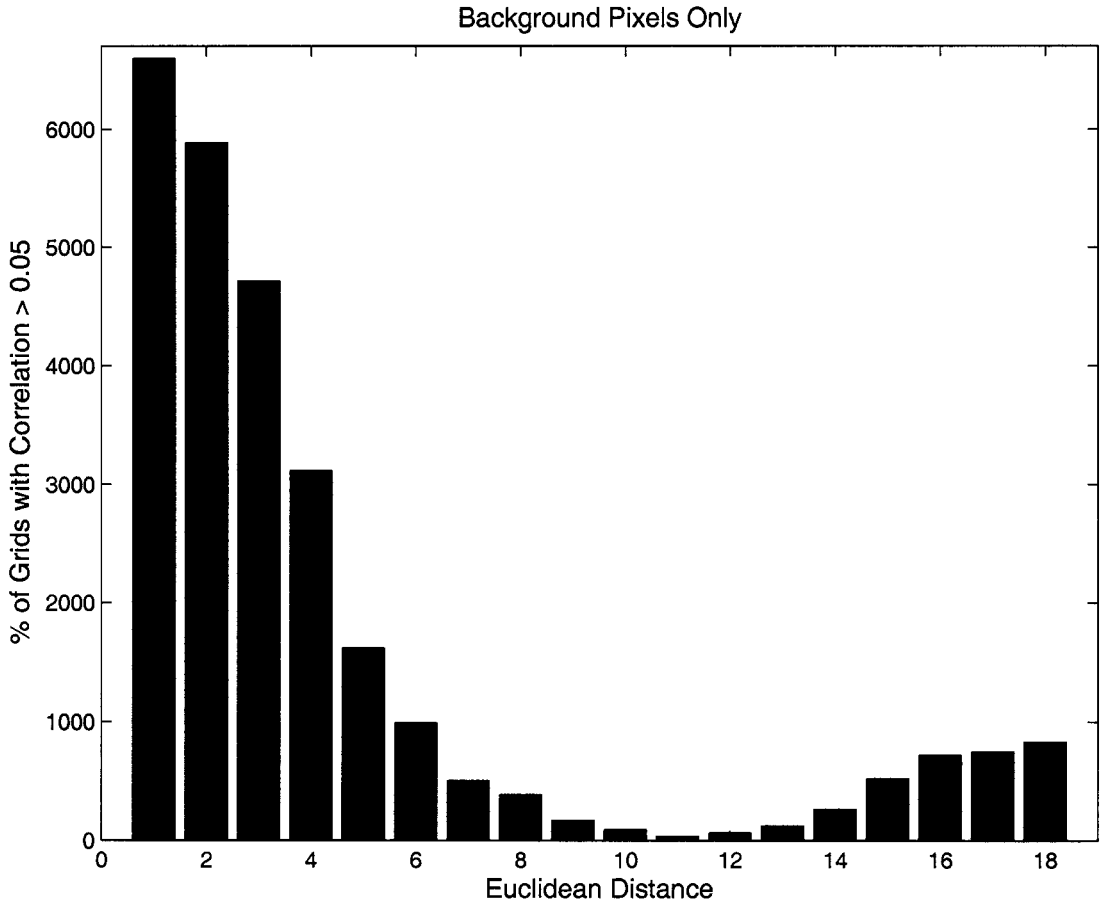


Figure 3.2: For pixels pairs at a given Euclidean distance *outside* spot regions, the number of spots with correlation above 0.05.

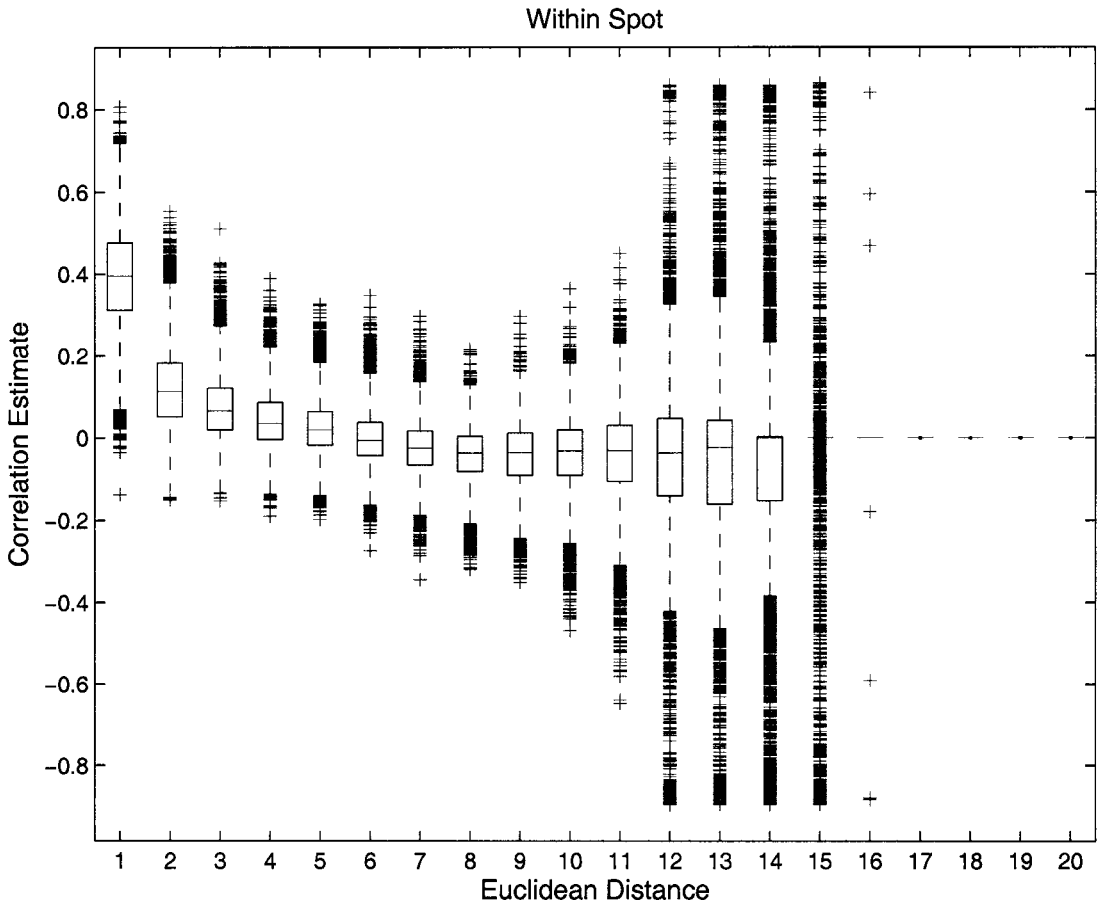


Figure 3.3: Correlation estimates for 6608 spots on a yeast array

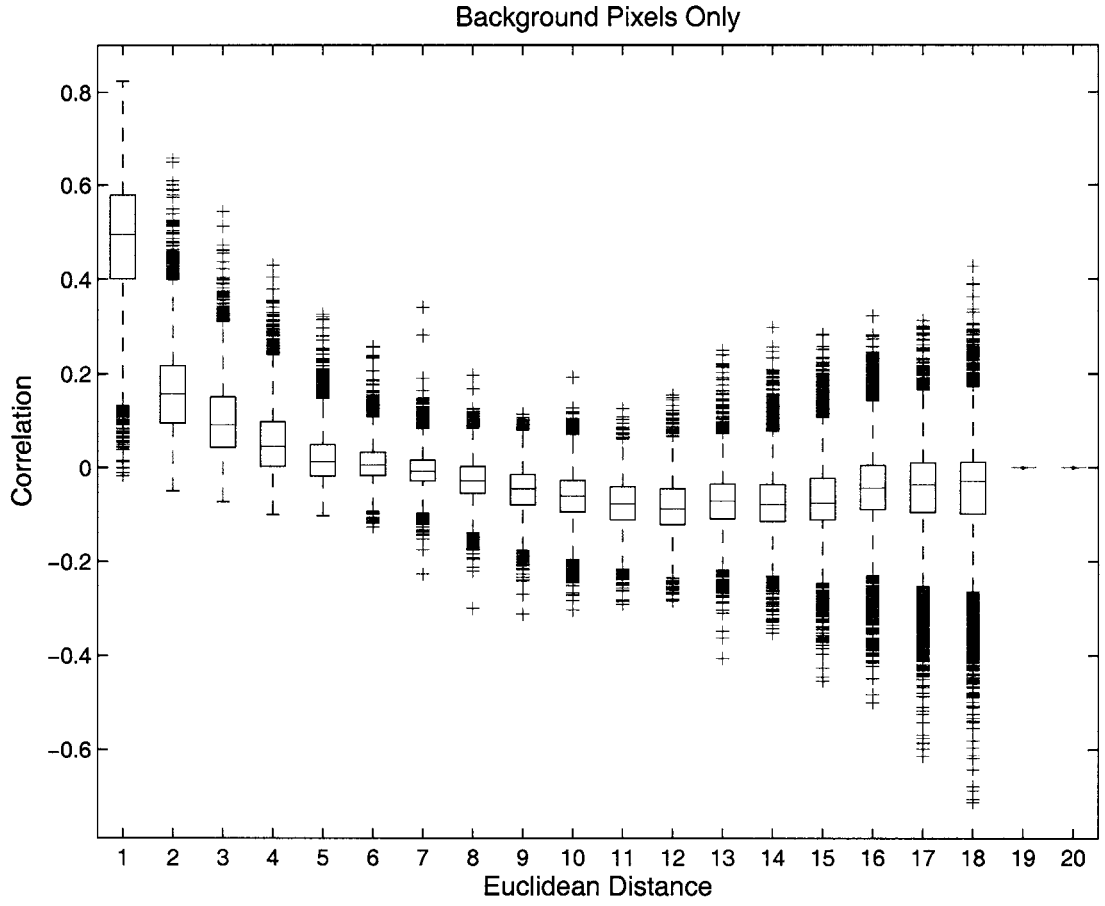


Figure 3.4: Correlation of local background values for 6608 spots on a yeast array

Table 3.2: Sandwich estimates for β_1 using the White-Domowitz weight. Data was simulated when $(\alpha_0, \alpha_1) = (0.8, 1.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 100.3$

	$\text{mean}(\hat{\beta}_0)$	1000.8	$\text{mean}(\hat{\beta}_1)$	4998.9
	95% CI for $\hat{\beta}_0$	(989.2,1012.4)	95% CI for $\hat{\beta}_1$	(4976.5,5021.3)
Distance B	1	2	3	4
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	46.702	69.062	85.755	90.214
95% CI for $\hat{se}(\hat{\beta}_1)$	(33.63,59.77)	(32.48,105.65)	(17.20,154.31)	(-10.37,190.80)
Distance B	5	6	7	8
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	82.951	68.245	56.054	47.951
95% CI for $\hat{se}(\hat{\beta}_1)$	(-39.94,205.84)	(-65.29,201.78)	(-76.70,188.81)	(-88.17,184.07)
Distance B	9	10		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	36.049	30.403		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-97.23,169.32)	(-122.22,183.03)		

standard error estimate over 200 simulations does not reach the expected value of 100.3 for any choice of B. When $B = 4$, the result is closest to the true value. The 95% coverage intervals indicate that standard error estimates vary widely from one simulation to the next and can even take on negative values. It should be noted that, except when $B = 1$, each interval does cover the true value.

In Table 3.3 when $(\alpha_0, \alpha_1) = (0.8, 1.0)$, the average standard error estimate over 200 simulations again does not reach the expected value of 118.588 for almost all choices of B. When $B = 4$, the result is close to the true value. Again, the 95% coverage intervals indicate that standard error estimates vary widely from one simulation to the next and can even take on negative values. Also, except when $B = 1$, each interval does cover the true value. Results for Table 3.4, where the correlation structure was exchangeable, are similar to conclusions from Tables 3.2 and 3.3 with one notable exception. That is, the ideal standard error estimate is achieved when $B = 1$ instead of $B = 4$.

General conclusions from these first three tables are that the variability of the estimates for (β_0, β_1) matched that expected for each data correlation structure simulated. The optimal distance, B, differs dependent on the type of correlation in the data, as expected. And finally, the White-Domowitz weight results in highly variable sandwich estimates.

Table 3.3: Sandwich estimates for β_1 using the White-Domowitz weight. Data was simulated when $(\alpha_0, \alpha_1) = (0.3, 1.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 118.588$

	$\text{mean}(\hat{\beta}_0)$	1001.0	$\text{mean}(\hat{\beta}_1)$	4998.7
	95% CI for $\hat{\beta}_0$	(987.2,1014.7)	95% CI for $\hat{\beta}_1$	(4972.3,5025.1)
Distance B	1	2	3	4
mean($\hat{Var}(\hat{\beta}_1)$)	46.196	83.502	112.426	121.947
95% CI for $\hat{se}(\hat{\beta}_1)$	(31.87,60.53)	(38.05,128.96)	(23.53,201.33)	(-11.85,255.74)
Distance B	5	6	7	8
mean($\hat{Var}(\hat{\beta}_1)$)	112.774	91.89	74.568	62.083
95% CI for $\hat{se}(\hat{\beta}_1)$	(-48.73,274.28)	(-80.27,264.06)	(-96.18,245.31)	(-111.95,236.12)
Distance B	9	10		
mean($\hat{Var}(\hat{\beta}_1)$)	47.026	41.012		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-128.93,222.99)	(-163.41,245.44)		

Table 3.4: Sandwich estimates for β_1 using the White-Domowitz weight. Data was simulated when $(\alpha_0, \alpha_1) = (1.5, 0.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 37.1$

	$\text{mean}(\hat{\beta}_0)$	996.2	$\text{mean}(\hat{\beta}_1)$	5000.3
	95% CI for $\hat{\beta}_0$	(947.3,1045.0)	95% CI for $\hat{\beta}_1$	(4988.3,5012.3)
Distance B	1	2	3	4
mean($\hat{Var}(\hat{\beta}_1)$)	36.986	35.857	34.148	32.566
95% CI for $\hat{se}(\hat{\beta}_1)$	(27.90,46.07)	(18.93,52.78)	(6.25,62.05)	(-7.34,72.47)
Distance B	5	6	7	8
mean($\hat{Var}(\hat{\beta}_1)$)	29.617	23.564	20.316	18.622
95% CI for $\hat{se}(\hat{\beta}_1)$	(-17.88,77.12)	(-29.67,76.80)	(-34.21,74.84)	(-36.60,73.84)
Distance B	9	10		
mean($\hat{Var}(\hat{\beta}_1)$)	15.251	12.83		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-43.69,74.19)	(-43.02,68.68)		

Table 3.5: Sandwich estimates for β_1 using the Newey-West weight. Data was simulated when $(\alpha_0, \alpha_1) = (0.8, 1.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 100.3$

	$\text{mean}(\hat{\beta}_0)$	1000.8	$\text{mean}(\hat{\beta}_1)$	4998.9
	95% CI for $\hat{\beta}_0$	(989.2,1012.4)	95% CI for $\hat{\beta}_1$	(4976.5,5021.3)
Distance B	1	2	3	4
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	46.702	57.825	69.259	75.147
95% CI for $\hat{se}(\hat{\beta}_1)$	(33.63,59.77)	(34.05,81.60)	(28.17,110.35)	(17.45,132.85)
Distance B	5	6	7	8
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	76.101	73.822	70.948	68.216
95% CI for $\hat{se}(\hat{\beta}_1)$	(6.90,145.30)	(-1.30,148.94)	(-7.14,149.03)	(-10.06,146.49)
Distance B	9	10		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	65.037	62.418		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-10.87,140.95)	(-13.00,137.84)		

Next, the same three simulations were run for the Newey-West estimate and are summarized in Tables 3.5, 3.6, and 3.7. In general, the conclusions are similar to those for the White-Domowitz weight. The range of standard errors generated from the Newey-West estimate is smaller than the White-Domowitz estimate, with far fewer negative values. When $B \geq 6$, the mean Newey-West standard error is larger and less biased than the White-Domowitz standard error for all correlation structures. For the exchangeable correlation structure summarized in Tables 3.4 and 3.7, the Newey-West estimate is closer to the truth for all values of B. In the case when an exponential decay correlation structure is simulated as in Tables 3.5 and 3.6, the Newey-West estimates are too small when $B < 6$.

Based on the comparison of the first two weight classes, neither type of estimate clearly outperforms the other on simulated microarray spot data. The Newey-West estimate seems more stable than the White-Domowitz estimate, but can sometimes incur more bias in the standard errors. And for both weight classes, there is no clear choice of B for all correlation structures. Hence, data will also be simulated for the WEAVE estimate that does not require a choice for B, but rather, a tuning constant C.

To construct the WEAVE estimate, a reasonable form for $\hat{\rho}$ must first be chosen. For microarray spot data, assuming a stationary structure, this correlation estimate is constructed

Table 3.6: Sandwich estimates for β_1 using the Newey-West weight. Data was simulated when $(\alpha_0, \alpha_1) = (0.3, 1.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 118.588$

	$\text{mean}(\hat{\beta}_0)$	1000.5	$\text{mean}(\hat{\beta}_1)$	4999.1
	95% CI for $\hat{\beta}_0$	(987.1,1013.9)	95% CI for $\hat{\beta}_1$	(4974.2,5024.0)
Distance B	1	2	3	4
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	46.144	64.283	82.967	93.845
95% CI for $\hat{se}(\hat{\beta}_1)$	(32.60,59.69)	(35.56,93.01)	(31.31,134.63)	(17.83,169.86)
Distance B	5	6	7	8
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	95.70	92.145	86.904	82.731
95% CI for $\hat{se}(\hat{\beta}_1)$	(3.70,187.70)	(-3.98,188.27)	(-8.39,182.20)	(-9.11,174.57)
Distance B	9	10		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	79.388	77.364		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-9.35,168.13)	(-10.58,165.31)		

Table 3.7: Sandwich estimates for β_1 using the Newey-West weight. Data was simulated when $(\alpha_0, \alpha_1) = (1.5, 0.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 37.1$

	$\text{mean}(\hat{\beta}_0)$	996.6	$\text{mean}(\hat{\beta}_1)$	5000.2
	95% CI for $\hat{\beta}_0$	(947.6,1045.6)	95% CI for $\hat{\beta}_1$	(4988.1,5012.3)
Distance B	1	2	3	4
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	37.021	36.439	35.664	34.657
95% CI for $\hat{se}(\hat{\beta}_1)$	(27.80,46.24)	(24.87,48.01)	(19.35,51.97)	(13.62,55.69)
Distance B	5	6	7	8
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	33.19	31.169	29.519	28.314
95% CI for $\hat{se}(\hat{\beta}_1)$	(9.30,57.08)	(5.24,57.09)	(3.37,55.66)	(1.57,55.06)
Distance B	9	10		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	27.029	25.932		
95% CI for $\hat{se}(\hat{\beta}_1)$	(1.04,53.02)	(0.84,51.03)		

such that $\rho_{ij,kl} = \rho_{i'j',k'l'}$ when $D_{ij,kl} = D_{i'j',k'l'}$. All unique pairs of pixels with a fixed Euclidean distance, B, are used to calculate the correlation coefficient. This is the same estimate Section 3.2.2 uses.

Tables 3.8, 3.9, and 3.10 show simulation results for the WEAVE estimate $w_{ij,kl} = 1 \wedge Cn\rho^2_{ij,kl}$. In all three cases, resulting sandwich estimates are seriously biased. And unlike the results in Lumley and Heagerty, these tables indicate worse performance than the Newey-West or White-Domowitz estimates. This is somewhat related to problems estimating $\hat{\rho}$ for a small number of pixels. To illustrate this, the true simulated values for ρ can be compared to their estimates. In the case when $(\alpha_0, \alpha_1) = (0.8, 1.0)$, the true values for ρ at Euclidean distances one through nine are

$$\rho = (0.1653, 0.0608, 0.0224, 0.0082, 0.003, 0.0011, 0.0004, 0.0002, 0.0001).$$

The average estimates of $\hat{\rho}$ over $N = 200$ simulations for these same distances are

$$\hat{\rho} = (0.1357, 0.0322, 0.0269, 0.0192, 0.0105, -0.0016, -0.0119, -0.0176, -0.0115).$$

Further, the choice of C impacts the WEAVE estimator just as the choice of B impacts the previous estimators. And the ideal value for C seems to vary from one simulation replicate to the next. It also appears that the bias in the WEAVE estimate is caused by more than just poor estimation of ρ. Bias still exists in simulation studies where $\hat{\rho}$ is replaced by the true value of ρ (results not shown).

3.3 Conclusions

In short, semi-parametric variance estimates do not describe microarray spot pixels accurately. A classic sandwich estimate requires replication over clusters. A spot consists of only one cluster and therefore the necessary replication does not exist. Weighted estimating equations require sufficient "pseudo-replication" so that the sandwich estimates approxi-

Table 3.8: Sandwich estimates for β_1 using WEAVEs. Data was simulated when $(\alpha_0, \alpha_1) = (0.8, 1.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 100.3$

	$\text{mean}(\hat{\beta}_0)$	1000.6	$\text{mean}(\hat{\beta}_1)$	4998.9
	95% CI for $\hat{\beta}_0$	(988.8,1012.5)	95% CI for $\hat{\beta}_1$	(4976.5,5021.2)
Constant C	0.1	0.25	0.5	
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	9.407	10.966	11.287	
95% CI for $\hat{se}(\hat{\beta}_1)$	(-15.11,33.93)	(-20.11,42.04)	(-24.70,47.27)	
Constant C	0.75	0.9		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	11.424	11.446		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-26.97,49.82)	(-28.51,51.41)		

Table 3.9: Sandwich estimates for β_1 using WEAVEs. Data was simulated when $(\alpha_0, \alpha_1) = (0.3, 1.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 118.588$

	$\text{mean}(\hat{\beta}_0)$	1000.8	$\text{mean}(\hat{\beta}_1)$	4999.2
	95% CI for $\hat{\beta}_0$	(987.5,1014.0)	95% CI for $\hat{\beta}_1$	(4973.7,5024.7)
Constant C	0.1	0.25	0.5	
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	14.634	17.517	18.316	
95% CI for $\hat{se}(\hat{\beta}_1)$	(-28.08,57.34)	(-30.25,65.28)	(-32.48,69.12)	
Constant C	0.75	0.9		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	19.624	20.209		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-35.18,74.43)	(-36.29,76.71)		

Table 3.10: Sandwich estimates for β_1 using WEAVEs. Data was simulated when $(\alpha_0, \alpha_1) = (1.5, 0.0)$. The true value for $\text{Var}(\hat{\beta}_1) = 37.1$

	$\text{mean}(\hat{\beta}_0)$	1003.5	$\text{mean}(\hat{\beta}_1)$	4999.7
	95% CI for $\hat{\beta}_0$	(950.7,1056.3)	95% CI for $\hat{\beta}_1$	(4987.2,5012.3)
Constant C	0.1	0.25	0.5	
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	4.464	5.551	6.101	
95% CI for $\hat{se}(\hat{\beta}_1)$	(-6.22,15.15)	(-8.39,19.49)	(-10.43,22.63)	
Constant C	0.75	0.9		
$\text{mean}(\hat{Var}(\hat{\beta}_1))$	6.033	5.896		
95% CI for $\hat{se}(\hat{\beta}_1)$	(-12.55,24.61)	(-13.45,25.24)		

mate the asymptotic distribution. Again, the pixel data within a spot are insufficient. If, as shown in Section 3.2.2, the lag autocorrelation is eight pixels, then there are only two "pseudo-replicates" within a data frame of 400 pixels. That is, with a lag of eight pixels, circle clusters are about the same size as microarray spots. Section 3.2.3 used a lag of five pixels in one case, resulting in only about five "pseudo-replicates". And as demonstrated in these simulations and in Lumley and Heagerty, this is insufficient data to provide accurate weighted sandwich estimators.

Chapter 4

GAUSSIAN MODELS AND PREDICTION ERROR MODELS

4.1 Introduction

Chapter 3 investigated the use of semi-parametric estimating equations to describe within spot signal variance. The investigation led to the conclusion that semi-parametric methods are not stable enough to handle correlated pixel data in microarray spots. Commonly, this conclusion leads to the use of a fully parametric approach or a non-parametric approach instead. In this chapter, both approaches will be outlined with an application that seems most suited to spot data. Each method tackles the problem with a different set of starting assumptions and a different resulting interpretation. In short, each approach will answer a slightly different question. The parametric approach will be discussed in Section 4.2. The non-parametric approach will be discussed in Section 4.3. The chapter will end with an analytic comparison of the two.

4.2 Gaussian Model for Spatial Correlation

A simple parametric model for pixel data allows for the incorporation of spatial correlation into moment estimates. If the distribution and spatial structure are fully specified, then parameters of interest are identifiable and solvable. But because the data lacks independent replication, the structure of the microarray spots must adhere to the distributional assumptions used in the model. It is doubtful that spot data conforms to any one definition of stochastic structure given the wide variety of technological biases in a microarray experiment. And hence, any parametric model is likely to be overly restrictive in its description of spot pixels. With this caveat in mind, the parametric assumptions are laid out below.

Because pixel intensities are basically continuous, we will assume that the process generating spot data is multivariate Gaussian. The spatial structure assumes decay with increasing distance. That is, assume $\vec{Y} \sim \text{MVN}(\vec{\mu}, \sigma^2\Sigma)$ where

$$\Sigma = \begin{pmatrix} 1 & \rho & \dots & \rho^{D_{ij,kl}} \\ \rho & 1 & \rho & \dots \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{D_{ij,kl}} & \dots & \rho & 1 \end{pmatrix}$$

and $\mu_{ij} = \beta_0 + \beta_1 \cdot I((i,j) \in S)$. Let $D_{ij,kl} = \max\{|i-j|, |k-l|\}$ as presented in Chapter 3. Under this model, the process is strictly stationary and isotropic. Second order *stationarity* means that the covariance of pixels pairs depends on their relative but not absolute positions. *Isotropy* implies that the covariance of pixel pairs depends only on the distance between positions. Because we are assuming a Gaussian model, we automatically have strict as opposed to weak stationarity and isotropy.

Although stationarity and a correlation decaying with distance are both reasonable assumptions, there are situations where this might not hold. A classic example is a spatial diffusion process spreading out from a fixed source. For a microarray example, spots with so-called "donuts" will have pixels pairs with increasing correlation over fixed distance. To generalize this example, if there is an error in the robotic printing process that generates systematic anomalies within spots, the variability will depend on absolute distance and not relative distance. Both of these scenarios will happen less as a microarray lab has more practice producing data.

The calculation of the within spot variance in the parametric framework is quite simple. For ease of presentation, assume that a spot signal is estimated by averaging over foreground pixels and subtract the average of background pixels,

$$\bar{S} = \frac{\sum_{i,j} Y_{ij} I((i,j) \in S)}{\sum_{i,j} I((i,j) \in S)} - \frac{\sum_{i,j} Y_{ij}(1 - I((i,j) \in S))}{\sum_{i,j} 1 - I((i,j) \in S)}.$$

If $N_1 = \sum_{i,j} I((i,j) \in S)$ and $N_0 = \sum_{i,j} 1 - I((i,j) \in S)$ and $c_{ij} = N_0 I((i,j) \in S) - N_1(1 - I((i,j) \in S))$, then $\bar{S} = \sum_{i,j}(c_{ij}Y_{ij})/N_0N_1$. Then the variance can be written as

$$\mathrm{Var}(\bar{S}) = \frac{1}{N_0^2 N_1^2}\mathrm{Var}\left(\sum_{i,j} c_{ij}Y_{ij}\right) = \frac{1}{N_0^2 N_1^2}\sigma^2\left\{\sum_{i,j} c_{ij}^2 + \sum_{i,j}\sum_{k,l} c_{ij}c_{kl}\mathrm{Cov}(Y_{ij}, Y_{kl})\right\}.$$

Under the Markovian model, $\mathrm{Cov}(Y_{ij}, Y_{kl}) = \rho^{D_{ij,kl}}$.

To solve for $\mathrm{Var}(\bar{S})$, a Newton-Raphson algorithm is used that calculates the optimal estimate for ρ using the correlation coefficient. $\hat{\mathrm{Var}}(\bar{S})$ yields a measure of within-spot variance that accounts for spatial correlation under a stationary Markovian process. Next are shown some preliminary examples of what the variance measure looks like for real data. Figures 4.1 and 4.2 compares different example spots and their standard error measures. Note that for each figure, the four spots shown are replicates that in theory, should have the same intensity level and spot quality. The variance estimates are taken on the log-ratio of the red and green channel images.

Figure 4.1 shows sample spots that are all of high intensity and Figure 4.2 shows spots of low intensity. The estimate of $\hat{\mathrm{Var}}(\bar{S})$ in Figure 4.1, decreases from spot A to spot B to spot D to spot C. The interpretation here is that quality increases from spots A to C and spot D is of lower quality than spot C. A corresponding calculation of the correlation for each distance between pixel pairs was also computed. Table 4.1 shows the correlation estimates for the spots in Figures 4.1 and 4.2.

For the spots in Figure 4.1, the correlation is largest for spot A. Spot D has correlation above zero when the pixel pair is one, but this rapidly drops to zero. Correlation in spot B does not rapidly decay. And spot C has the lowest correlation level. Hence the higher variance estimate of spots A and B are likely related to their high correlation values. But the higher variance of spots A and B may also be due to a strong signal intensity. This indicates the possibility of a mean-variance relationship, resulting in larger variances for spots of greater intensity or larger size. Because the variance estimate is taken on the

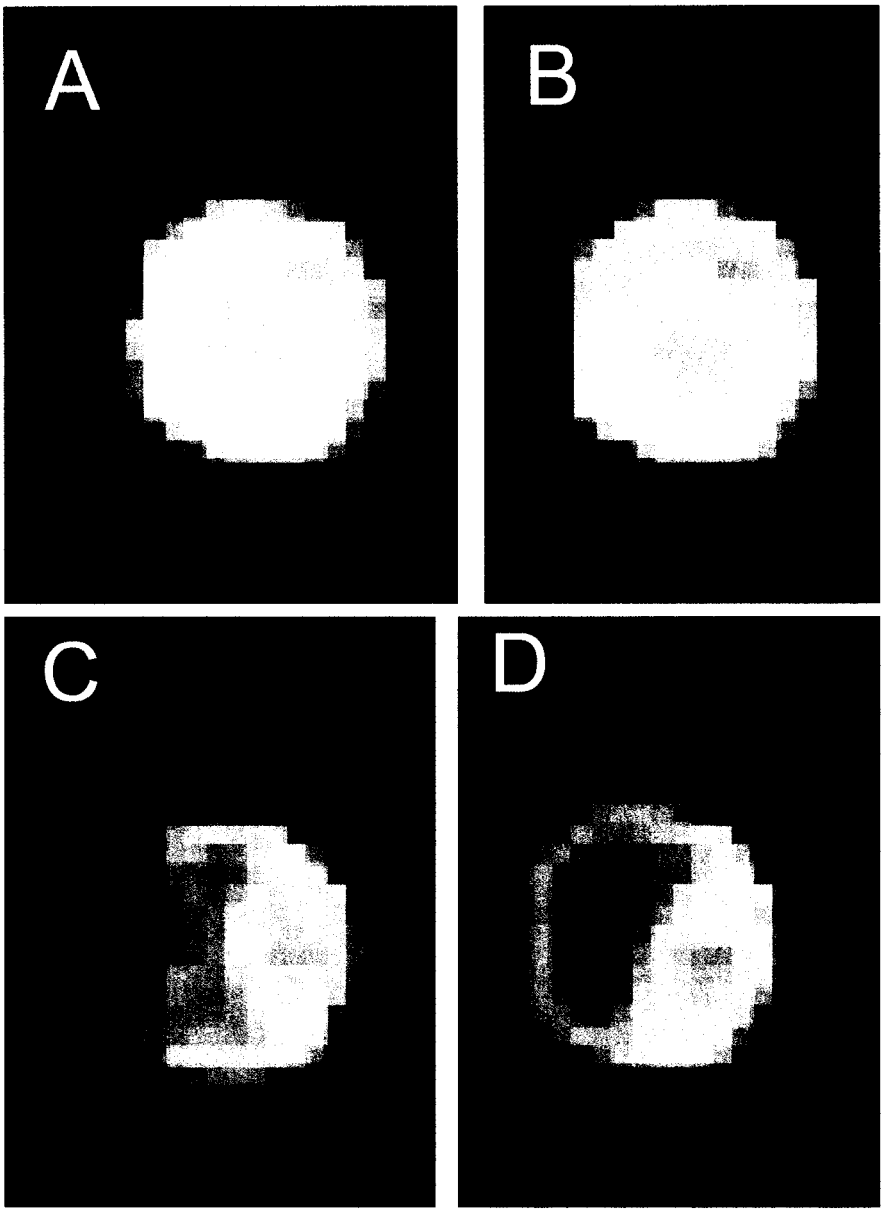


Figure 4.1: Standard error estimates for each sample spot are A: 0.165, B: 0.135, C: 0.119, and D: 0.129

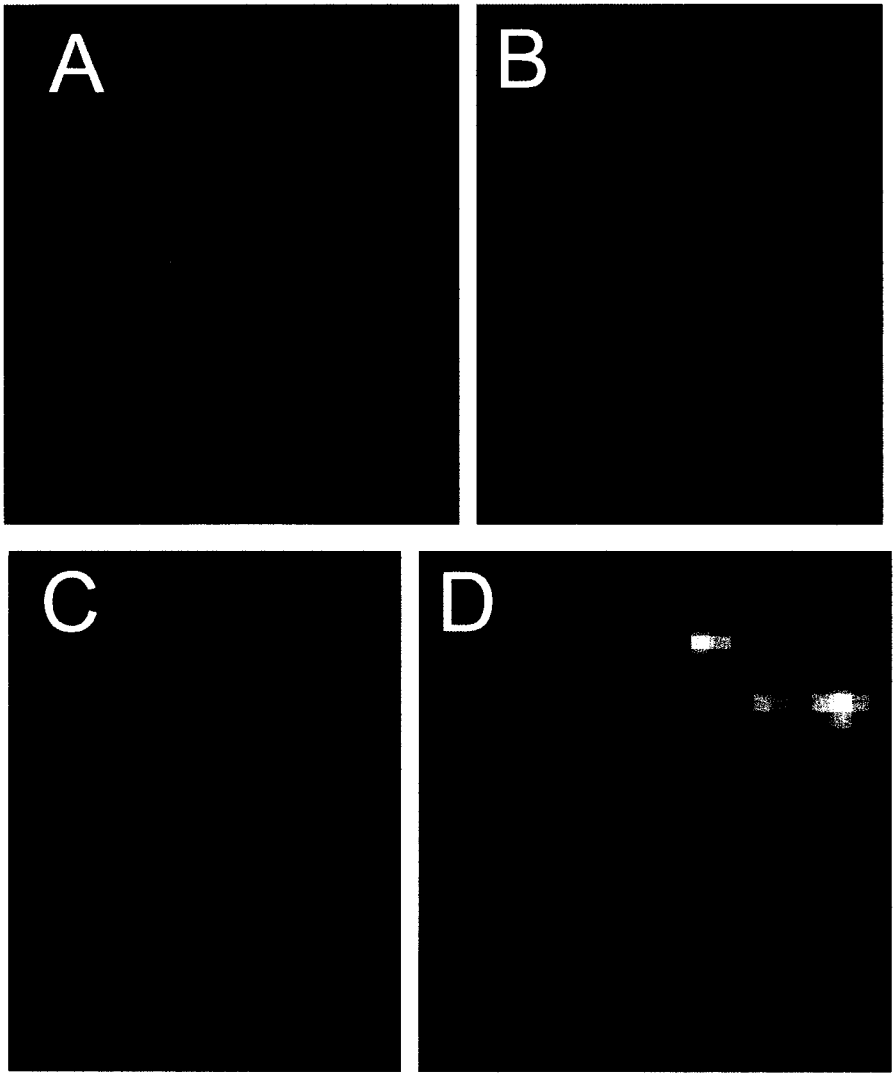


Figure 4.2: Standard error estimates for each sample spot are A: 0.129, B: 0.135, C: 0.135, and D: 0.114

Table 4.1: Correlation of pixel pairs for spots in Figures 4.1 and 4.2.

Distance	1	2	3	4	5	6	7	8
Spot A	0.2916	0.0744	0.0264	0.0141	-0.0072	-0.0166	-0.0279	-0.0203
Spot B	0.1488	0.0572	0.0826	-0.0016	-0.0285	0.0022	-0.0341	-0.0337
Spot C	0.1219	0.0157	0.0355	0.0448	-0.0336	-0.0123	0.0448	-0.0429
Spot D	0.2311	0.0013	-0.0112	0.0023	-0.0213	-0.0616	-0.0484	0.0009
Spot A	0.0993	0.093	0.0773	0.0542	-0.0141	0.0176	-0.0319	-0.0116
Spot B	0.3933	0.1561	-0.0023	-0.0174	-0.0167	-0.007	-0.058	-0.0209
Spot C	0.2367	0.0436	0.029	0.027	-0.0342	0.0175	-0.0393	-0.0379
Spot D	0.2387	0.0399	-0.0232	-0.0347	0.0134	-0.0017	0.0057	-0.0004
Array average	0.1953	0.062	0.0485	0.0339	0.0206	0.0099	-0.0013	-0.007

log-ratio of the channels, this should diminish any possible mean-variance association. A possible mean-variance relationship will be investigated further in Chapter 5.

The estimates for $\hat{\text{Var}}(\bar{S})$ in Figure 4.2 are smallest for spot D and then increasing from spot A to C. This implies that spot D has the highest quality of the four spots. A subjective inspection of these spots is not likely to assign spot D the highest quality. The author's personal interpretation, although it may differ from other researchers, is that spot D has the worst quality of the four. The correlation estimates in Table 4.1 again show some association with the variance estimates. Spot D has the smallest variance, but not the smallest correlation. Spot B definitely has the strongest correlation values but does not have a significantly larger variance estimate. This shows that the correlation in the over-dispersion factor may not be the only important influence on quality and other factors such as random noise or signal intensity may also play a role.

4.3 Prediction Error Model for Signal Quality Measurement

Clearly, a fully parametric model to describe spot variability involves some significant assumptions about the nature of spot pixels and their relationship with each other. The advantage of the parametric model is that it is easily interpretable and fully identifiable. This is particularly useful for incorporation into downstream analysis of microarray data.

An example might be to use the spot variance measure in a variance components model that tests for differentially expressed genes. The idea here is that variability would be deconstructed into components for the microarray, block level, gene level and the spot level with both a variance and spatial correlation component.

The goal in this section will be to develop a non-parametric variability measure that requires fewer assumptions while minimizing later loss of generality. That is, the non-parametric measure should be easily incorporated into downstream analysis for differentially expressed genes or classification models. This measure can not be used in a variance components model as will be clear later in this section. Namely, the non-parametric statistic will be a prediction error model instead of a variance estimate.

Here, a moment is taken to describe again what exactly the goal of estimating spot variability is. The goal is to describe the *quality* of the spot and its *reliability* for use in analysis. If the goal is to quantify reliability, this does not necessarily correspond to a variance measure as described in Section 4.2. For example, if there is high correlation between spot pixels, an over-dispersion measure will yield a high variance measure. Low correlation between spots, or strictly random noise will yield a lower variance measure. But spots with high correlation between pixels may be more uniform, having more similar values and therefore possibly being of higher quality. Spots with low correlation will be less uniform and potentially of lower quality. Simulations will investigate this rationale further in the next section.

If spots with more uniformity are penalized with larger variance measures, then perhaps a variance measure is not the goal after all. The goal is to develop a measure of spot reliability that can be used meaningfully. In this spirit, a metric that compares each pixel to its surrounding pixels is proposed. If a pixel has the same intensity as its *nearest neighbors*, then the spot is of consistent quality. And so, the mean square predication error (MSPE) is introduced. A discussion of expected prediction error can be found in [26]. In general the

MSPE looks like

$$\text{MSPE} = \sqrt{\sum_{(i,j)\in S} \left(Y_{ij} - s_{ij}(\vec{X})\right)^2 / N_1}$$

where s_{ij} is a smoothed function of pixels surrounding Y_{ij}. This smoother should aptly describe the relationship between $\vec{X}$ and $\vec{Y}$. The form of the smoother could be a spline smooth, a wavelet, a loess, etc.

For the purposes of describing spot data, this work proposes $s_{ij}(\vec{X}) = \sum_{(i',j')}\{Y_{(i',j')} : D_{ij,i'j'} = 1\}/8$. This is commonly known as a nearest-neighbor function. The general form of this function is $s_{ij}(\vec{X}) = \sum_{(i',j')}\{Y_{(i',j')} : (i',j') \in N_k(i,j)\}$ where $N_k(i,j)$ can be of any size, but for this research a neighborhood size of one is used. The MSPE is similar to a spatial semi-variogram with a lag of one in each direction. For a lattice of size $n \times n$, and lags of k and l and the horizontal and vertical directions, the semi-variogram is

$$\hat{\gamma}(k,l) = \sum_{i=1,\cdots,n-k} \sum_{j=1,\cdots,n-l} (Y_{i,j} - Y_{i+k,j+l})^2 / 2(n-k)(n-l).$$

The semi-variogram is optimal when Y is normally distributed [23].

Although the MSPE is a non-parametric measure of quality, there remain assumptions about the structure of the data that using the MSPE imposes. The largest assumption is that, of the pixels surrounding a data point Y_{ij}, only the immediate circle of pixels is important. That is, if a pixel pair has a distance $D_{ij,kl} > 1$ it does not have a predictive relationship. Naturally, the MSPE can be extended to include larger neighborhood sizes. Here the neighborhood size of one was chosen because microarray spots are small with a limited number of pixel pairs. The MSPE also assumes a weakly stationary and isotropic process.

Although the MSPE is similar to a variance estimate, the interpretation of values is different. As is learned in introductory statistics, $\text{Var}(X) = E(X - E(X))^2$. The $E(X)$ is estimated by the method of moments or a maximum likelihood estimate. In the case of prediction error, $E(X)$ is replaced with a smoother, $s(X)$. Smoothers partition the data

into smaller pieces and thus yield piecewise estimates of the first moment. Thus, the MSPE is really examining the *consistency* of the pixel data instead of its variability in the classic sense.

Here, this section revisits the sample spots in Figures 4.1 and 4.2. The MSPE for the spots in Figure 4.1 are A: 0.763, B: 0.6398, C: 0.627, and D: 0.716. Note that the error estimate decreases from spots A, D, B and spot C has the smallest prediction error. This result differs from that of Section 4.2 as spot D is given a higher error estimate.

The MSPE for the spots in Figure 4.2 are A: 0.658, B: 0.6389, C: 0.621, and D: 0.743. This indicates that the direction of reliability is from worst to best, spot D, A, B, and C. This trend is in complete opposition to the Markovian error model. Whereas in Section 4.2 spot C is the "worst" spot and D is the "best". Here, spot D is the "worst" spot and C is the "best". And, the MSPE values for the eight spots do not necessarily correspond to the pattern of empirical correlation estimates seen in Table 4.1. Again, other factors than correlation play a role in prediction error estimation.

Because comparisons of spots in Figures 4.1 and 4.2 for various quality measures based on subjective assessments is insufficient, the next section will compare quality measures on simulated data. Chapter 5 will be devoted to using objective assessments on real data to compare the non-parametric and parametric models.

4.4 Analytic Comparisons

If microarray spot data conforms to the normality assumptions given in Section 4.2, it is of interest to see how the MSPE performs under normality assumptions. This section compares the expectations of quality measures for normal data with a two-dimensional autoregressive covariance structure.

4.4.1 MSPE under Normality Assumptions

For simplicity of presentation, let it first be assumed here that the data of interest is a time series. We will then extend results to two dimensions. Suppose the data of interest is a time series that looks like $Y_1, Y_2, \ldots, Y_n$. Further, assume that this time series follows the normal distribution given in Section 4.2. Under these conditions, we are interested in the expected value of the MSPE. For the time series,

$$\text{MSPE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \frac{Y_{i-1} + Y_{i+1}}{2})^2$$

and we wish to determine E[MSPE].

Let $\vec{a}$ represent a vector such that $\vec{a}'\vec{Y} = Y_i - \frac{Y_{i-1}+Y_{i+1}}{2}$. That is,
$\vec{a} =< 0, \ldots, 0, -\frac{1}{2}, 1, -\frac{1}{2}, 0, \ldots, 0 >$ where the value 1 is in the i^{th} position. Recall that $\text{Var}(\vec{Y}) = \Sigma$ has elements $\sigma^2\rho^{|i-j|}$. Without loss of generality, we assume that the $E(\vec{Y}) = \vec{0}$. If $\vec{Y} \sim N(\vec{0}, \Sigma)$, then $\vec{a}'\vec{Y} \sim N(\vec{0}, \vec{a}'\Sigma\vec{a})$ and $\text{E[MSPE]} = E(\vec{a}'\vec{Y})^2 = \vec{a}'\Sigma\vec{a}$. And so assuming normality, our expectation of interest has the following form:

$$\begin{aligned}
\text{E[MSPE]} &= \vec{a}'\Sigma\vec{a} \\
&= \sigma^2\left[-\frac{1}{2}\left(-\frac{1}{2} + \rho - \frac{1}{2}\rho^2\right) + \left(-\frac{1}{2}\rho + 1 - \frac{1}{2}\rho\right) - \frac{1}{2}\left(-\frac{1}{2}\rho^2 + \rho - \frac{1}{2}\right)\right] \\
&= \sigma^2\frac{(\rho-3)(\rho-1)}{2}.
\end{aligned}$$

This can be compared with the $\text{Var}(\vec{Y}) = \frac{\sigma^2}{n^2}\left[n + 2\sum_{i=1}^{n-1}(n-i)\rho^i\right]$, the variance estimate from Section 4.2, not accounting for background subtraction. There are two major differences between E[MSPE] and $\text{Var}(\vec{Y})$. The first difference is that the incorporation of correlation terms in the E[MSPE] is multiplicative and in the $\text{Var}(\vec{Y})$ is additive. That is, E[MSPE] looks like something of form $\sigma^2 * f(\rho)$ and $\text{Var}(\vec{Y})$ looks like something of the form $\sigma^2 + f(\rho)$. The second difference is that E[MSPE] only has two exponential forms for correlation, ρ and ρ^2. This is in keeping with the assumption that only the nearest

Table 4.2: Values of spot quality measures for time series data.

n	ρ	$\text{Var}(\vec{Y})$	E[MSPE]
5	0.1	0.2346	1.305
10	0.1	0.1198	1.305
15	0.1	0.0804	1.305
5	0.5	0.445	0.625
10	0.5	0.26	0.625
15	0.5	0.1822	0.625
5	0.8	0.7243	0.22
10	0.8	0.5429	0.22
15	0.8	0.4285	0.22
20	0.8	0.3512	0.22

neighbors are used to calculate spatial correlation. This contrasts with the $\text{Var}(\vec{Y})$ where terms as small as ρ^{n-1} are included.

For given values of ρ and series size n, there can be quite a difference in the values of E[MSPE] and $\text{Var}(\vec{Y})$. Table 4.2 displays these differences for various parameter values. This table shows two not surprising facts: (1) that MSPE increases with decreasing correlation and (2) that the $\text{Var}(\vec{Y})$ decreases with increasing series size and increases with increasing correlation.

It is fairly straightforward to extend results to the case of spatial data. Suppose now that the data of interest is a vector of image pixels that looks like

$$\vec{Y} = Y_{11}, Y_{12}, \ldots, Y_{1n}, Y_{21}, Y_{22}, \ldots, Y_{2n}, \ldots, Y_{n1}, Y_{n2}, \ldots, Y_{nn}.$$

Again, assume that this image follows the normal distribution as given in Section 4.2. Under these conditions, we are interested in the expected value of the MSPE. For two dimensions,

$$\text{MSPE} = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(Y_{ij} - \frac{\sum_{(i',j')}\{Y_{(i',j')} : D_{ij,i'j'} = 1\}}{8})^2$$

and we wish to determine E[MSPE].

Now, let $\vec{a}$ represent a vector such that $\vec{a}'\vec{Y} = Y_{ij} - \frac{\sum_{(i',j')}\{Y_{(i',j')}:D_{ij,i'j'}=1\}}{8}$. That is,

$$\vec{a} = \begin{cases} 0 & \text{for pixel } ij \\ -\frac{1}{8} & \text{for pixels } (i,j-1)(i,j+1)(i-1,j-1)(i-1,j)(i-1,j+1) \\ & (i+1,j-1)(i+1,j)(i+1,j+1) \\ 0 & \text{otherwise.} \end{cases}$$

Recall that $\text{Var}(\vec{Y}) = \Sigma$ has elements $\sigma^2\rho^{\max\{|i-k|,|j-l|\}}$. Like the time series example, $\text{E[MSPE]} = E(\vec{a}'\vec{Y})^2 = \vec{a}'\Sigma\vec{a}$. And so assuming normality, our expectation of interest has the following form:

$$\begin{aligned}
\text{E[MSPE]} &= \vec{a}'\Sigma\vec{a} \\
&= \sigma^2 \cdot < \rho^{\max\{|i-k|,|j-l|\}} - \frac{1}{8}\rho^{\max\{|i-k|,|j-1-l|\}} - \frac{1}{8}\rho^{\max\{|i-k|,|j+1-l|\}} \\
&\quad -\frac{1}{8}\rho^{\max\{|i-1-k|,|j-1-l|\}} - \frac{1}{8}\rho^{\max\{|i-1-k|,|j-l|\}} - \frac{1}{8}\rho^{\max\{|i-1-k|,|j+1-l|\}} \\
&\quad -\frac{1}{8}\rho^{\max\{|i+1-k|,|j-1-l|\}} - \frac{1}{8}\rho^{\max\{|i+1-k|,|j-l|\}} \\
&\quad -\frac{1}{8}\rho^{\max\{|i+1-k|,|j+1-l|\}} : \forall k,l > \cdot\vec{a} \\
&= \sigma^2\left[(1-\rho) - \frac{1}{8}(\frac{\rho}{2} - \frac{1}{8} - \frac{3\rho^2}{8}) - \frac{1}{8}(\frac{\rho}{2} - \frac{1}{8} - \frac{3\rho^2}{8}) - \frac{1}{8}(\frac{3\rho}{4} - \frac{1}{8} - \frac{5\rho^2}{8})\right. \\
&\quad -\frac{1}{8}(\frac{\rho}{2} - \frac{1}{8} - \frac{3\rho^2}{8}) - \frac{1}{8}(\frac{3\rho}{4} - \frac{1}{8} - \frac{5\rho^2}{8}) - \frac{1}{8}(\frac{3\rho}{4} - \frac{1}{8} - \frac{5\rho^2}{8}) \\
&\quad \left. -\frac{1}{8}(\frac{\rho}{2} - \frac{1}{8} - \frac{3\rho^2}{8}) - \frac{1}{8}(\frac{3\rho}{4} - \frac{1}{8} - \frac{5\rho^2}{8})\right] \\
&= \sigma^2\left(\frac{9}{8} - \frac{13\rho}{8} + \frac{\rho^2}{2}\right).
\end{aligned}$$

This can be compared with the $\text{Var}(\vec{Y}) = \frac{\sigma^2}{n^4}\left[n^2 + \sum_{i,j,k,l=1}^{n}\rho^{\max\{|i-k|,|j-l|\}}\right]$, the variance estimate from Section 4.2, not accounting for background subtraction. Figure 4.3 shows the E[MSPE] and $\text{Var}(\vec{Y})$ for values of ρ and several image sizes. Clearly, the MSPE will decrease with increasing correlation. And the normal model that accounts for spatial correlation with an over-dispersion factor, will (1) increase with increasing values of ρ and

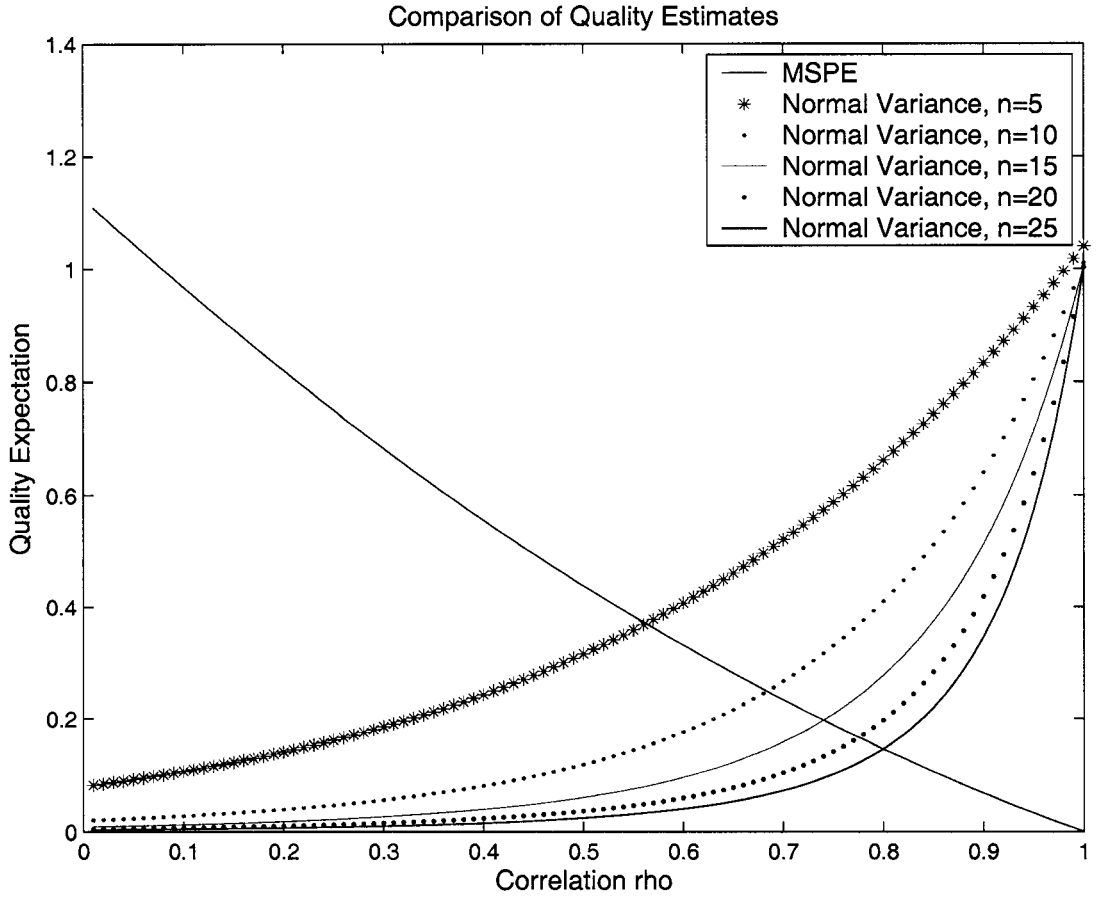


Figure 4.3: Comparison of E[MSPE] and Var($\vec{Y}$) for several image sizes. Here the number of pixels in the image is n^2.

(2) decrease with increasing image sizes.

The above calculations do not account for pixels on the boundary of images. The following derivations provide a more general way to calculate E[MSPE] accounting for boundary pixels. First, suppose that a vector $Y_1, Y_2, \ldots, Y_n$ follows a normal distribution, $\vec{Y} \sim N(\vec{0}, \Sigma)$ as in Section 4.2. Then perform a Cholesky decomposition that yields a upper triangular matrix $\mathbf{A}$ such that $\mathbf{A}'\mathbf{A} = \Sigma$. Further consider a vector $\vec{a_{ij}}$ such that

$$\vec{a_{ij}'}\vec{Y} = Y_{ij} - \frac{\sum_{D_{ij,i'j'}=1} Y_{i'j'}}{\sum_{i',j'} I\left(D_{ij,i'j'} = 1\right)}.$$

Let vector $\vec{b}_{ij} = A\vec{a}_{ij}$. Then,

$$\text{E[MSPE]} = \frac{1}{n^2} \sum_{i,j=1}^{n} \vec{b}'_{ij}\vec{b}_{ij}$$

and in the absence of boundaries, this value reduces to $\left(\frac{9}{8} - \frac{13\rho}{8} + \frac{\rho^2}{2}\right)$. In the presence of boundary conditions, the E[MSPE] is always larger.

Once again, the E[MSPE] is compared with $\text{Var}(\vec{Y}) = \frac{\sigma^2}{n^4}\left[n^2 + \sum_{i,j,k,l=1}^{n} \rho^{\max\{|i-k|,|j-l|\}}\right]$ for spots following the specified normal distribution. Figure 4.4 shows the E[MSPE] and $\text{Var}(\vec{Y})$ for values of ρ and several image sizes. The MSPE will decrease with increasing correlation and does not vary much with image size. And the normal model that accounts for spatial correlation with an over-dispersion factor, will increase with increasing values of ρ and decrease with increasing image size.

To further illustrate the spot quality measures presented, the chapter concludes with the performance of these measures on simulated spot data. Figures 4.5 through 4.8 provide three dimensional topologies of the sample simulated spots for four different values of the vector (ρ, σ^2). The aim here is to look at spot quality measures for spots of

1. low correlation and low random noise

2. mid-level correlation and low random noise

3. high correlation and low random noise

4. high correlation and high random noise.

Spots, as in Chapter 3, are simulated from a multivariate normal distribution, $\vec{Y} \sim N(\vec{\mu}, \Sigma)$ where $\mu = \beta_0 + \beta_1 I(\mathbf{S})$. Here, $\mathbf{S}$ is a circle of diameter 10 and the entire image has a width and length of 20 pixels. Spots were simulated 100 times and the sample MSPE and standard error $\hat{\text{Var}}(\bar{S})$ were calculated for each. Here the quality measures are standard

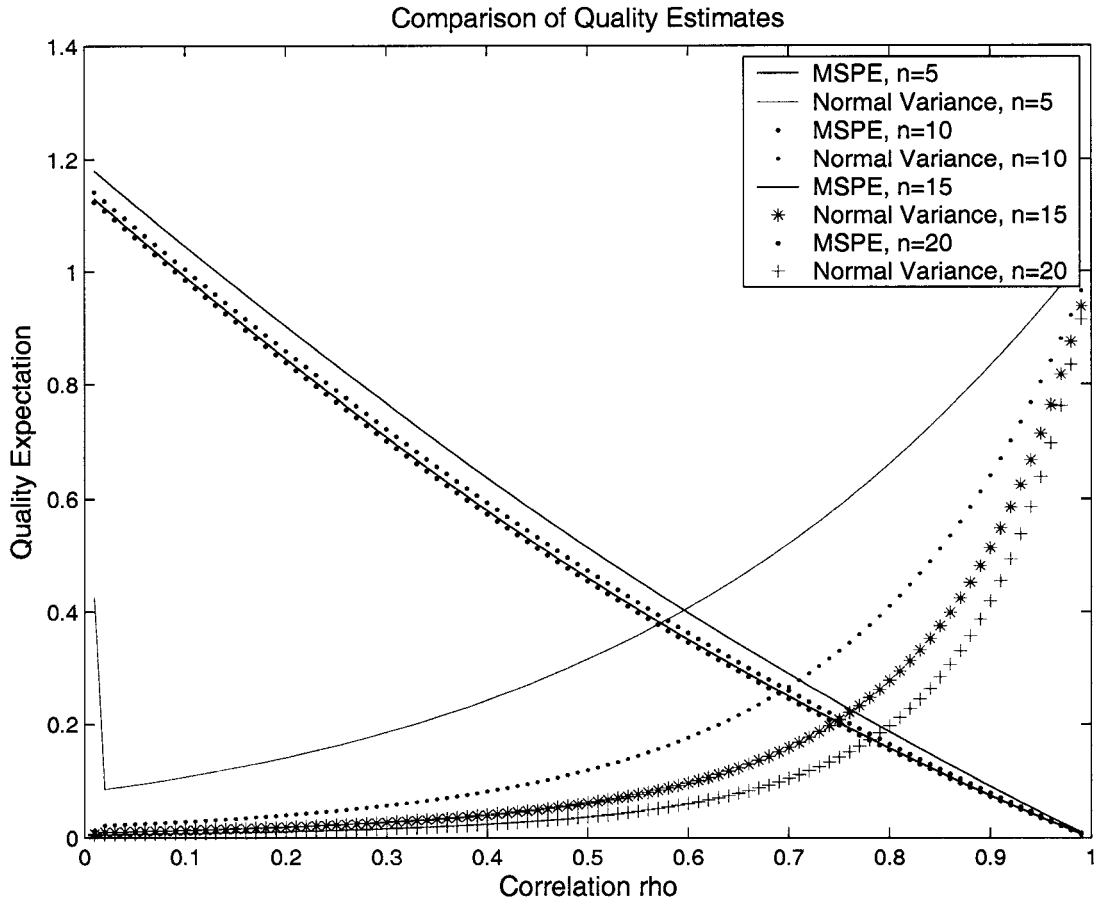


Figure 4.4: Comparison of E[MSPE], accounting for boundary pixels, and $\mathrm{Var}(\vec{Y})$ for several image sizes. Here the number of pixels in the image is n^2.

Table 4.3: Summary of spot quality measures for 100 simulated spot images.

ρ	σ^2	mean MSPE	MSPE 95% CI	mean Normal SE	Normal SE 95% CI
0.1	10^3	31.375	(29.10,33.65)	5.104	(4.03,6.18)
0.5	10^3	21.404	(19.88,22.93)	9.774	(5.99,13.55)
0.8	10^3	12.625	(11.23,14.02)	9.833 [1]	(8.28,11.26)
0.8	10^4	39.892	(35.84,43.95)	32.35 [2]	(27.10,37.72)

errors as given in Sections 4.2 and 4.3, not the variance estimates. The results are shown in Table 4.3.

Table 4.3 shows values that match up well with the expected values derived and shown in Figure 4.4. That is, for $\rho = 0.1$, the expected value is $\sqrt{10^3 \mathrm{E}[\mathrm{MSPE}]} = 31.38$. This is very close to simulation results. The same is true for other parameter values, for $\rho = 0.5$, $\sqrt{10^3 \mathrm{E}[\mathrm{MSPE}]} = 21.32$, for $\rho = 0.8$, $\sqrt{10^3 \mathrm{E}[\mathrm{MSPE}]} = 12.41$ and $\sqrt{10^4 \mathrm{E}[\mathrm{MSPE}]} = 39.26$.

A comment should be made about the missing values generated in Table 4.3. The Markovian model did not always yield a measurable quantity. 40% of the time, the estimate failed when the correlation, ρ, was set to 0.8. The estimation fails when the value of $\hat{\rho}$ is negative. This indicates less stability in this model for increasing correlation levels. Caution should be exercised when using the Newton-Raphson for the Markovian model in the presence of high correlation.

Figures 4.7 and 4.8 show that spots with large amounts of correlation need not be penalized unless coupled with a large random noise component. That is, the spot in Figure 4.7 is of high quality while the spot in Figure 4.8 is not. Variance estimates with over-dispersion factors for spatial correlation will penalize both spots more heavily than those with less correlation. The MSPE does not penalize these spots equally and in fact gives the spot in Figure 4.7 the smallest error rate of any of the four examples.

[2] 40% of data from these simulations yielded missing values due to negative correlation estimates in the Newton-Raphson algorithm solution. Hence the median and inter-quartile range are used to summarize normal model estimates.

[2] 40% of data from these simulations yielded missing values due to negative correlation estimates in the Newton-Raphson algorithm solution. Hence the median and inter-quartile range are used to summarize normal model estimates.

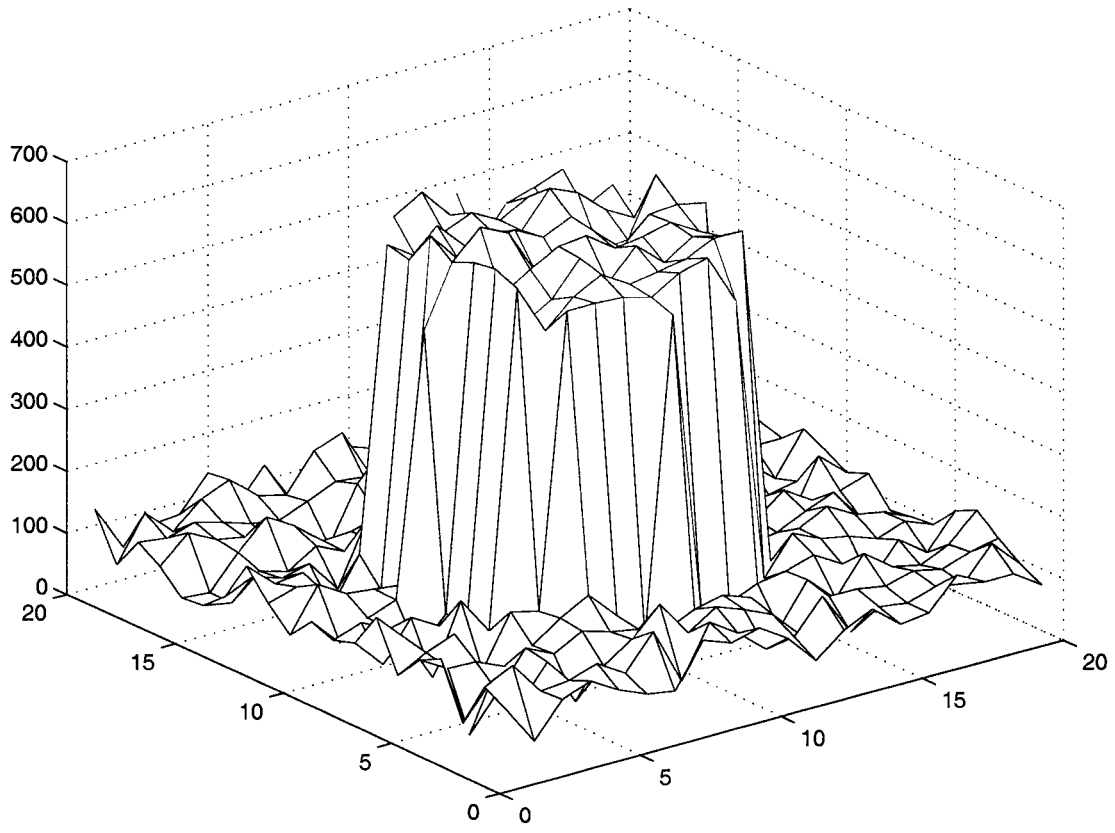


Figure 4.5: Spot simulated with parameters $\rho = 0.1$, $\mu = 500$, and $\sigma^2 = 1000$

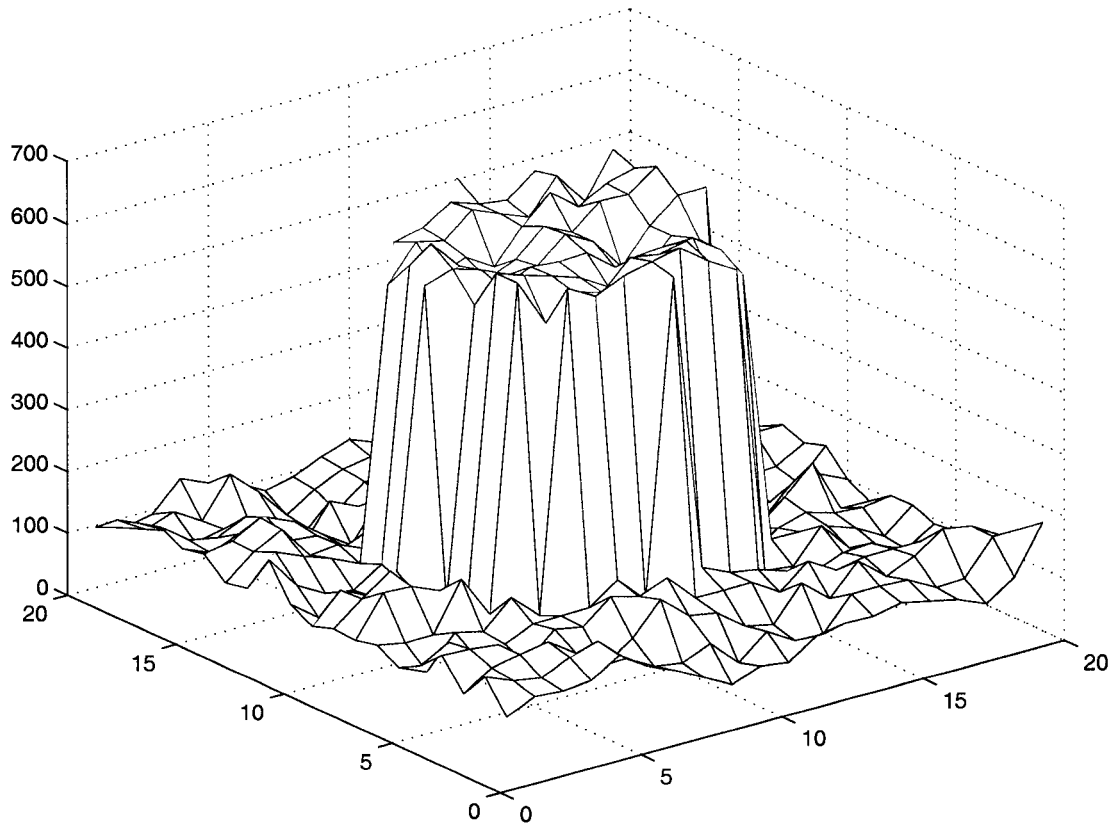


Figure 4.6: Spot simulated with parameters $\rho = 0.5$, $\mu = 500$, and $\sigma^2 = 1000$

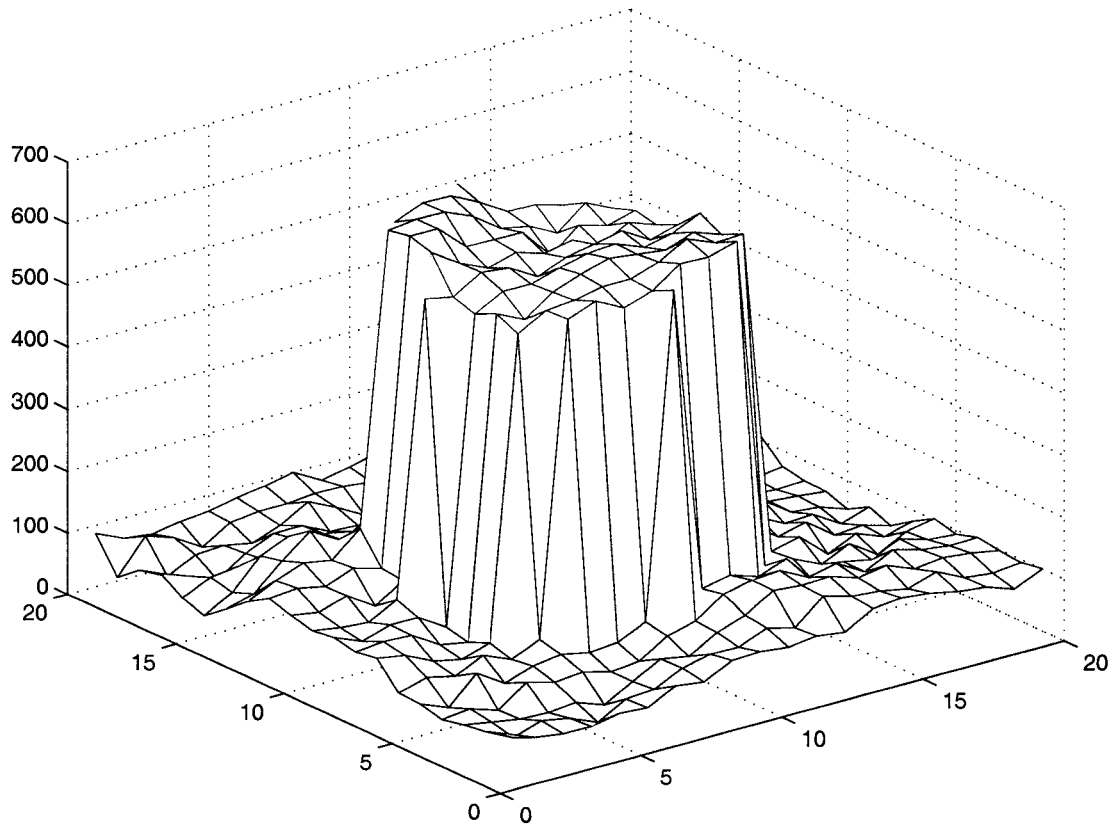


Figure 4.7: Spot simulated with parameters $\rho = 0.8$, $\mu = 500$, and $\sigma^2 = 1000$

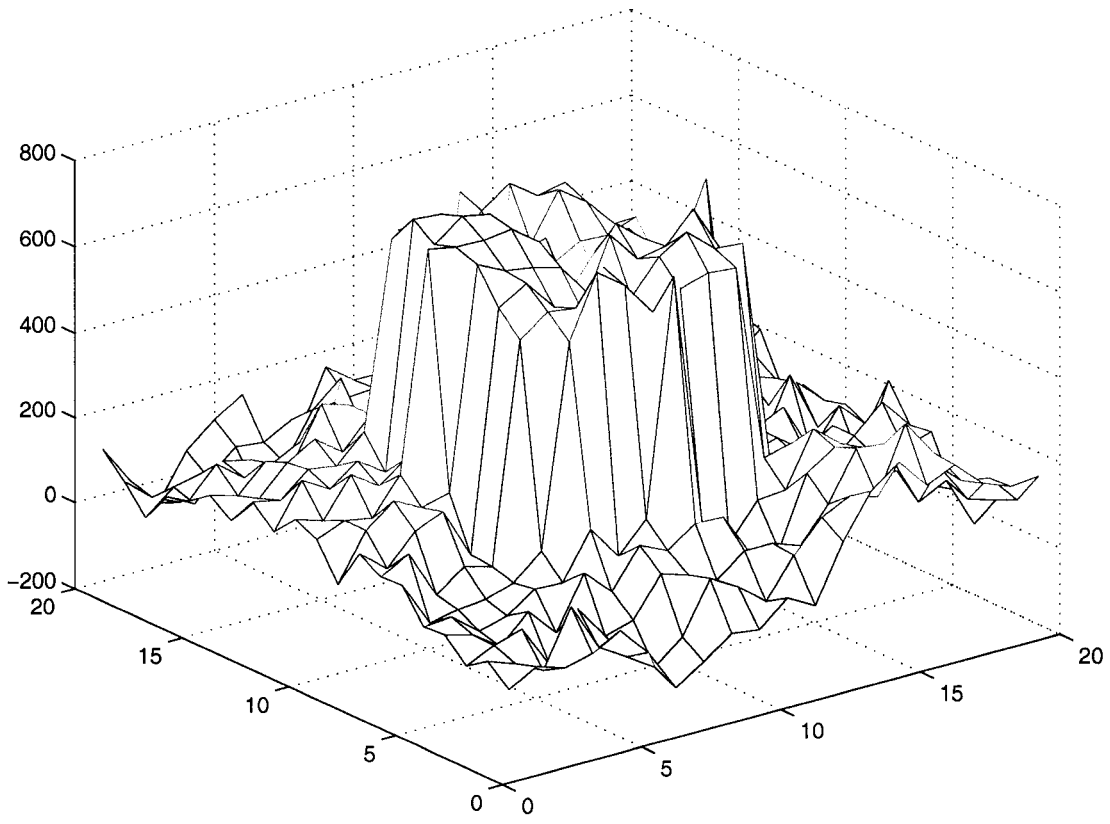


Figure 4.8: Spot simulated with parameters $\rho = 0.8$, $\mu = 500$, and $\sigma^2 = 10000$

4.4.2 Conclusions

This chapter introduced two spot quality scores, a parametric estimate based on variance from a multivariate normal distribution and a non-parametric estimated based on a summary of prediction error. The assumptions and interpretation of both estimates are quite different. The common ground is that both situations assume a spatial correlation that decays with distance and is dependent on relative and not absolute distance. The cursory comparison of the two scores on eight sample spots indicated that neither quality estimate completely matched the author's subjective judgment of the best and worst spots. The comparison also showed that while correlation is important in the quality scores, it does not play the only role and this is not surprising. Certainly other variations such as random noise contribute to the scores.

Analytic and simulation studies later in the chapter indicate that, under a normal distribution assumption, both scores are entirely dependent on three factors: image size, noise level, and correlation level. Real data clearly does not follow the rigorous assumptions of the normal model and will have more than three factors contributing to spot quality. The MSPE, as a more flexible measure is better equipped to deal with these other factors. The next chapter investigates the performance of spot quality scores in real data sets.

Chapter 5

PERFORMANCE OF PREDICTION ERROR AND GAUSSIAN MODEL ESTIMATES

This chapter will examine the ability of spot quality measurements to identify poor spots in real data examples. The focus will be on the two quality estimates introduced in Chapter 4. The first measure introduced was a parametric estimate of variance based on a Markovian model and will be referred to in this chapter as σ_{spot}. The second measure introduced was a non-parametric estimate of prediction error and will be referred to as the MSPE.

The Fred Hutchinson Cancer Research Center Microarray facility headed by Jeff Delrow has graciously provided all of the images and GenePix output that will be used in this chapter. All of the microarrays used yeast clones hybridized to mRNA extracted from yeast cells. There are two datasets included in this chapter, each with different designs and goals. For all experiments, red and green channels represent the same yeast strain meaning that the resulting ratio is known.

The first dataset (Dataset One) was planned as part of the Public Health Sciences Shared Resource grant to extensively investigate the sources of variation in microarray experiments. Four dye swap experiments investigate the role of dye incorporation on resulting data. Eight dilution experiments investigate the consistency of ratios for differing concentrations of mRNA extracts. Both extracts are from the same yeast strain. The dilution experiments also incorporate a dye swap and two replicates within each experiment. Table 5.1 displays the design of these eight dilution experiments.

Each of 96 yeast open reading frame (ORF) clones in Dataset One are replicated 64 times, for a total of 6,144 spots. That is, within each of the 16 array blocks, each ORF is replicated

Table 5.1: Experimental Design of Dataset One

Concentration ratio	Replicates	Dye Swap
100% versus 75%	2	Yes
100% versus 50%	2	Yes

four times. Every block within the array is a replicate. This allows for researchers to examine within-block-between-spot variation as well as between-block-within-spot variation. The clone set used for each dilution and dye swap experiment is the same.

The second dataset (Dataset Two) consists only of a single microarray experiment comparing wild type yeast to itself for 6,608 different cDNA clones. This array was visually inspected by array lab personnel for spot quality. Spots subjectively considered to be of poor quality were assigned a GenePix flag of -100. GenePix also assigns flags to this data when their spot detection routine fails and these flags take on values of either -50 or -75.

5.1 Mean-Variance Relationship

Chapter 4 pointed out a possible relationship between spot quality estimates and spot signal estimates, or, a mean-variance relationship. To investigate this on a broader scale, the MSPE and σ_{spot} are calculated for every spot on one array in Dataset One. First the estimates are found for one channel on the array only (the red channel) and then for the log-ratio of the two channels, $\log_2 R/G$. The log-ratio is taken at the pixel level before calculating quality estimates. Figures 5.1 and 5.2 show the results. The first plot, for the red channel only, shows a definite relationship between the spot mean and quality estimates. But the second plot, on the log-ratio, shows that this relationship has largely disappeared. Thus, if spot quality estimates are derived from pixel level log-ratio values, there is no association with the spot signal.

In Figure 5.2, a slight relationship between signal and spot quality still exists. For the MSPE measurement, larger ratios with values greater than roughly 0.5 have larger MSPE values. This plot also points out a possible asymmetry in the ratio values. Even if these

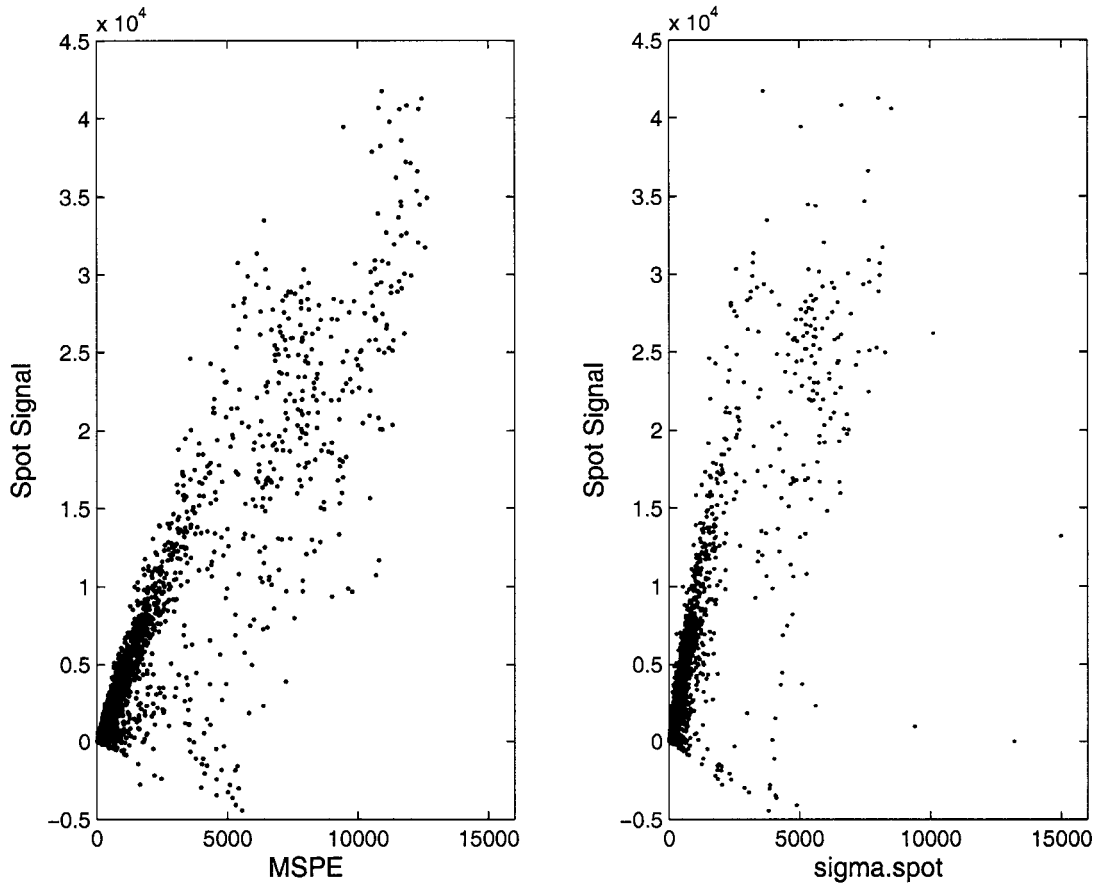


Figure 5.1: Comparison of spot mean and standard error measurements for MSPE and σ_{spot}. The error measures use pixel data from the red channel only. This plot does not include 62 extreme values of σ_{spot} greater than 15,000.

values are normalized to zero, there seem to be more large positive ratio values than large negative ratio values. And, since these large positive ratio values also have greater MSPE scores, it seems that these spots might be of lesser quality. A similar phenomenon appears in the plot of σ_{spot}. It is reassuring that spot quality scores capture this asymmetry.

The same investigation can be applied to determine the relationship between spot size and spot quality estimates. Again, estimates on the red channel alone and on the log-ratio combination were used. The number of pixels in a spot represents the spot size. Using boxplots of quality estimates for each possible spot size, we can examine their relationship

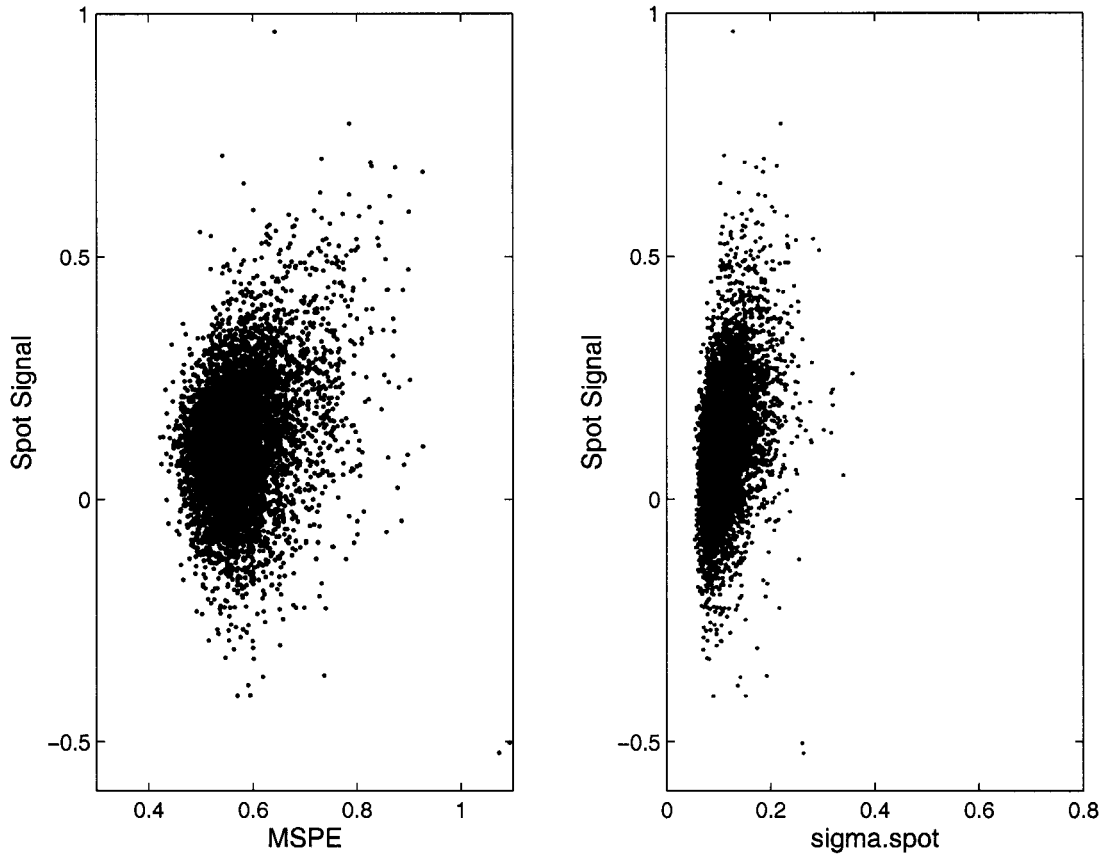


Figure 5.2: Comparison of spot mean and standard error measurements for MSPE and σ_{spot}. The error measures use pixel data from the log-ratio of the red and green channels.

in Figures 5.3 and 5.4. For quality estimates on the red channel alone, large values are more often assigned to very large spots. This corresponds to a peak in the boxplot at a spot size of 137 pixels. Spots of this size have much larger error estimates, possibly indicating spots of poor quality at this size level.

When looking at quality estimates from the ratio of channels some of the values at the 137 pixels point diminish. But, there still seems to be higher error measures for this particular spot size. The most probable reason for this is that SignalViewer flags spots and assigns default circles of 137 pixels in area to these regions. So, what is really observed here is a relationship between quality estimates and SignalViewer flagging.

Finally, when looking at the values of σ_{spot} from the ratio of channels, there seems to be decreasing standard error with spot size. As the expected value of σ_{spot} depends on size, the pattern seen in the lower graph of Figure 5.4 matches this expectation. In general, however, the graphs in this section demonstrate very little relationship between either of the two spot quality estimates and spot signal or size, when using the pixel-level ratio of channel intensities. The graphs also indicate a connection between quality estimates and SignalViewer flagging. This will be examined further in the next section of this chapter.

5.2 Dataset One

In this section, the performance of MSPE and σ_{spot} will be compared on Dataset One. The aims of this section are as follows:

1. Examine the ability of these two measures to predict flags given in SignalViewer and GenePix,

2. Compare spot quality with the consistency of replicate spots,

3. Assess the relative importance of spot quality compared with other sources of variability on microarrays.

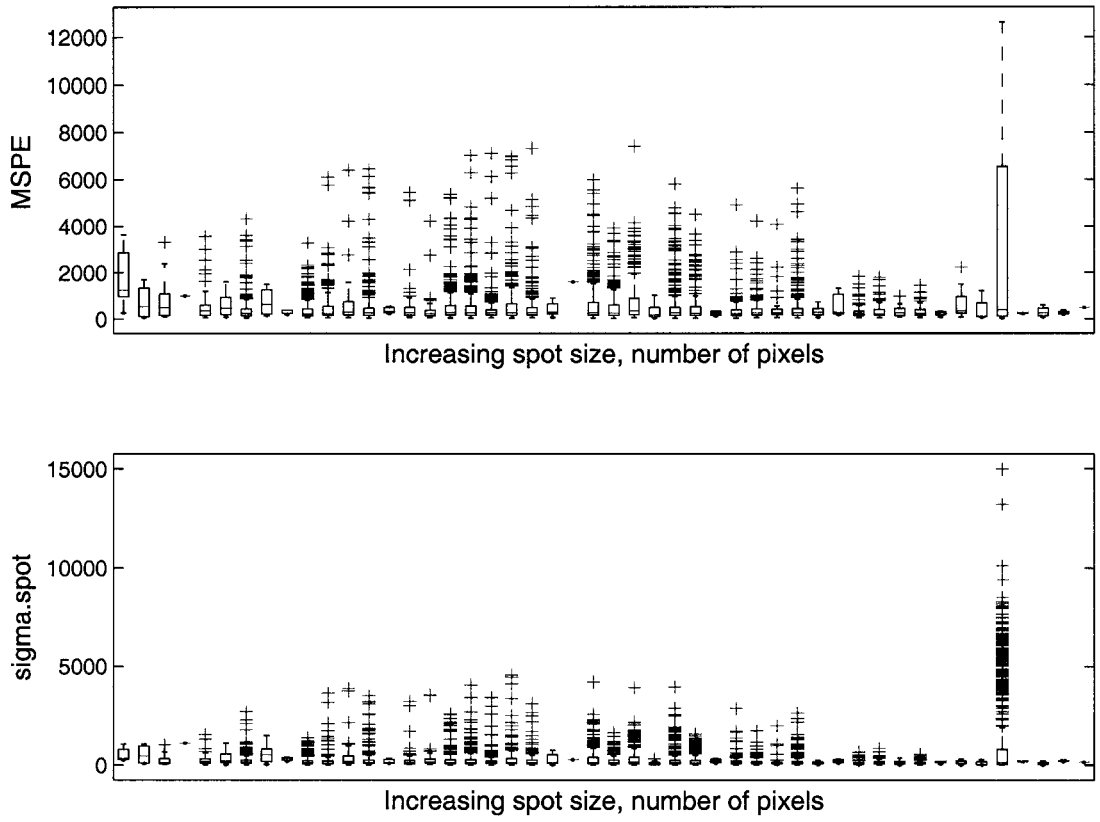


Figure 5.3: Comparison of spot size and standard error measurements for MSPE and σ_{spot}. The error measures use pixel data from the red channel only. Spot size in number of pixels is plotted in increasing order. This plot does not include 62 extreme values of σ_{spot} greater than 15,000.

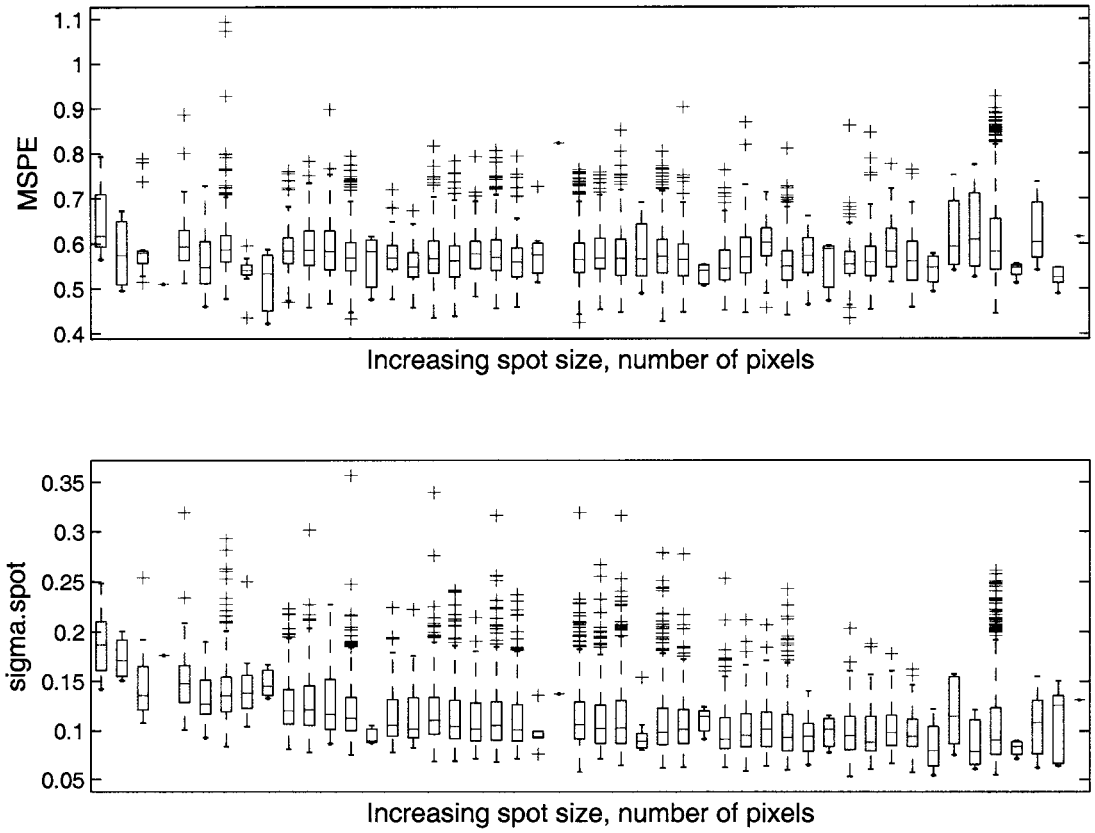


Figure 5.4: Comparison of spot size and standard error measurements for MSPE and σ_{spot}. The error measures use pixel data from the log-ratio of the red and green channels. Spot size in number of pixels is plotted in increasing order.

5.2.1 SignalViewer and GenePix flags

Implementing methods for spot flagging given in Chapter 2, estimates of spot quality can be assessed on flagged spots in Dataset One to determine the relationship between these scores. Good flagging mechanisms should align well with good spot quality scores.

Using a series of boxplots, four figures will examine the relationship between spot quality and spot flags for the four experiments comparing 50% concentrations to 100% concentrations within Dataset One. Figure 5.5 gives the relationship between the MSPE and SignalViewer flags. These boxplots show that the average MSPE for artifact noise flags and flags indicating no spot detection is always elevated compared to the MSPE for spots with no flag. Sometimes the MSPE score for the irregular ellipse flag is also elevated. Flags for low expression have MSPE values very similar to those spots with no flag. Thus, MSPE values line up very well with flag categories. Figure 5.6 gives the association between MSPE and GenePix flags. On average, only 4% of spots are flagged by GenePix in this dataset and even fewer have been manually flagged. The experiments comparing 100% to 50% concentration show elevated MSPE values only for manually flagged spots. The last experiment comparing the dye swap, 50% to 100% concentration shows *decreased* values of MSPE for flagged data. The cause of this trend will be explored further in a later part of this section.

Figure 5.7 shows the relationship between σ_{spot} and SignalViewer flags. These boxplots do not have the same pattern as shown for in Figure 5.5 for the MSPE scores. Rather, spots flagged by SignalViewer tend to have *lower* σ_{spot} values. This is certainly counterintuitive and indicates that using the parametric model results in poor prediction of SignalViewer flagging mechanisms. The exception here is the boxplot of the first replicate in the 50% to 100% comparison. In this exceptional case, average σ_{spot} values are larger for spots with SignalViewer flags indicating poor quality. Figure 5.8 gives the association between σ_{spot} and GenePix flags. The experiments comparing 100% to 50% concentration show slightly elevated σ_{spot} values for all flagged spots. The last two experiments comparing the dye

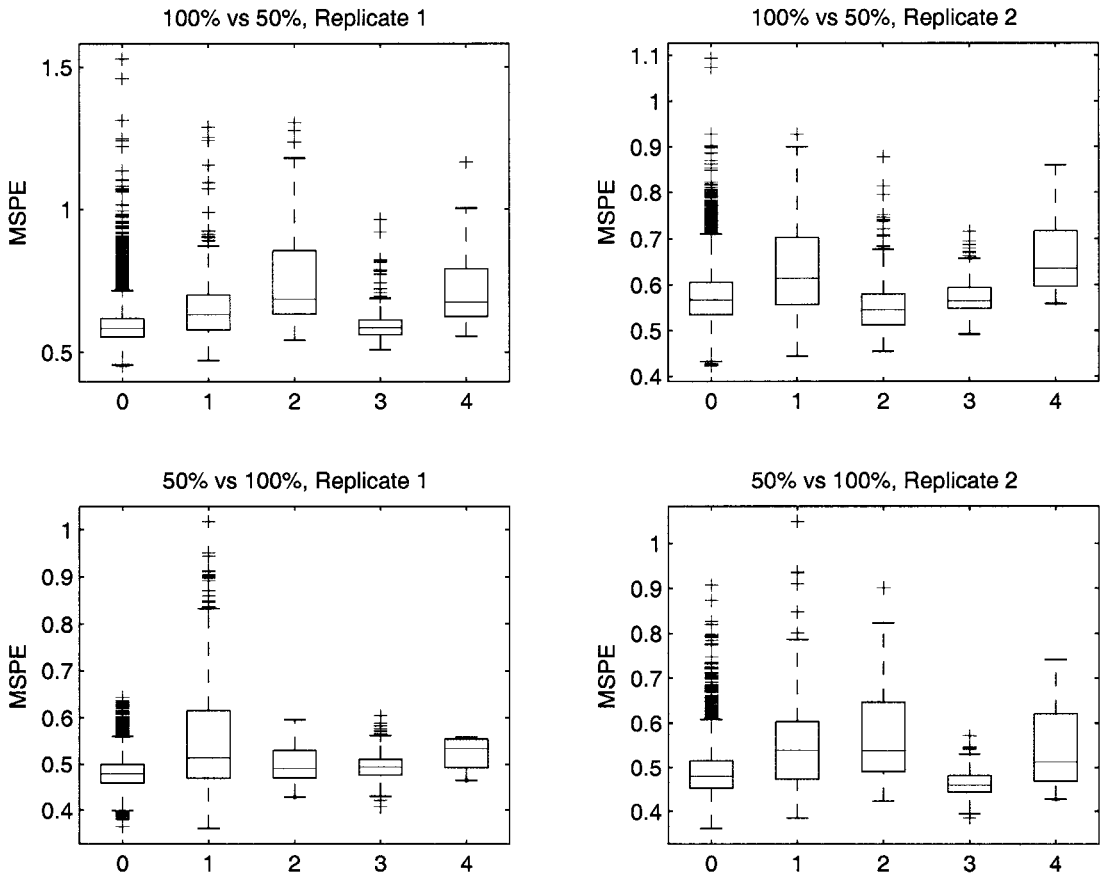


Figure 5.5: For each experiment, the range of MSPE values for each type of SignalViewer flag. The SignalViewer flags are coded as 0=no flag, 1=artifact noise, 2=irregular ellipse, 3=low expression, and 4=no spot detection.

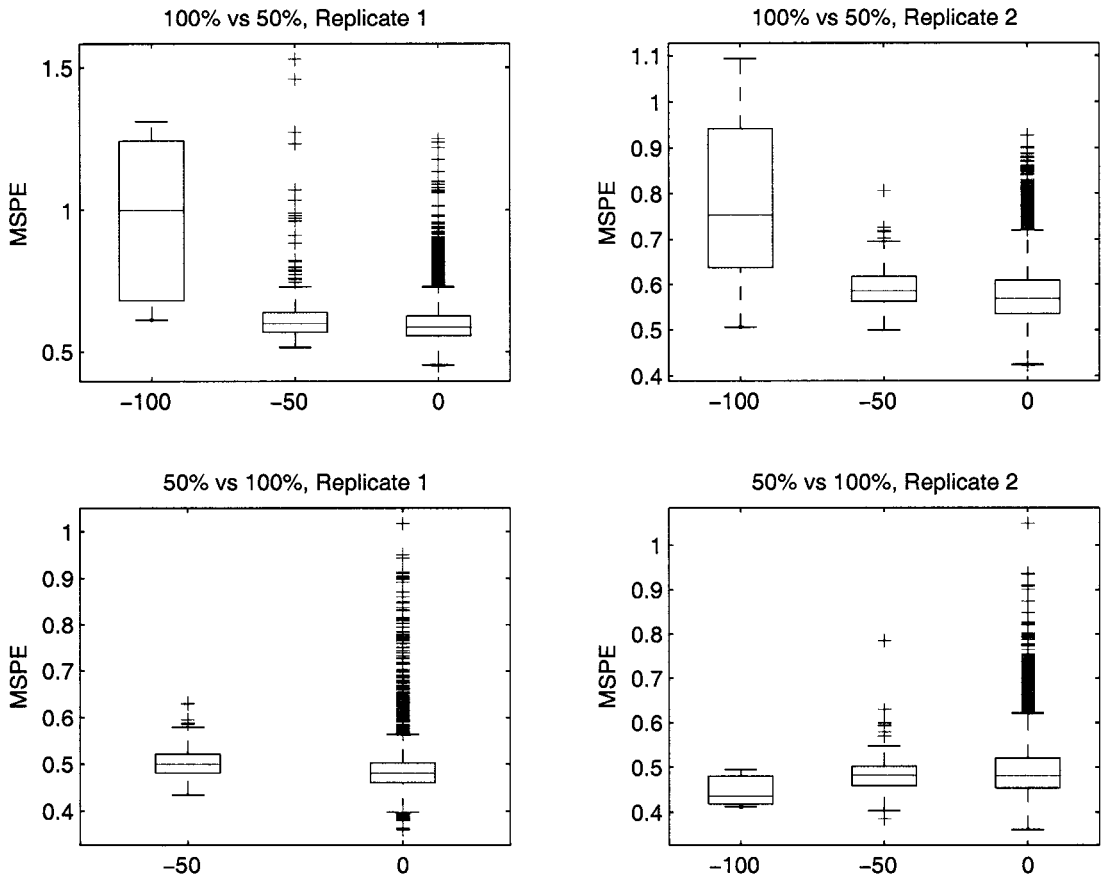


Figure 5.6: For each experiment, the range of MSPE values for each type of GenePix flag. The flags for GenePix are coded as 0=no flag, -50=no spot detection, and -100=manual flag.

swap, 50% to 100% concentration again shows *decreased* values of σ_{spot} for flagged data.

It is of interest to determine exactly why spot quality measures are lower for spots with GenePix flags in the last two experiments. To do this, the focus will be on the fourth experiment, the second replicate of the 50% to 100% comparison. In this experiment, only three spots have manual flags (GenePix values of -100) out of 6144 total spots. 153 spots have automated GenePix flags, that are assigned when the GenePix spot detection routine fails. Figures 5.9 and 5.10 show the same sample block within this experiment and the corresponding GenePix and SignalViewer flags. This first thing to notice within this sample block is the interference of noise in the lower right hand corner of the block. Measurements in this corner will clearly be unreliable as they are unduly influenced by artificial noise. The GenePix flags in Figure 5.9 have green boxes highlighting automatic flags and one white box with an indicating arrow for the single manually assigned flag. GenePix seems to choose spots of low expression for flagging, although this is not consistent throughout the block. It is unclear why these spots are flagged over other examples in the block as there are other spots with little signal that go unflagged. Those spots in the lower right corner that are clearly unreliable have not been flagged by GenePix. This can be contrasted with the SignalViewer flags, indicated by red circles in Figure 5.10. The SignalViewer flags pick up most of the spots in the lower right hand corner as well as most of the spots with low expression levels.

MSPE values for the spots in the lower right corner of this sample block are certainly larger than the average score of 0.5, with values roughly between 0.7 and 0.9 (results not shown). The σ_{spot} values are not much larger than normal for these spots. Further, given the poor assignment of GenePix flags, both automated and manual, for this block, it is unlikely that the GenePix flags provide a good surrogate of spot quality. Hence, the lack of association between MSPE, σ_{spot}, and GenePix flags is not surprising. But, the positive association between MSPE scores and SignalViewer flags, along with their consistent performance in Figure 5.10, indicate that both of these measures can capture actual spots

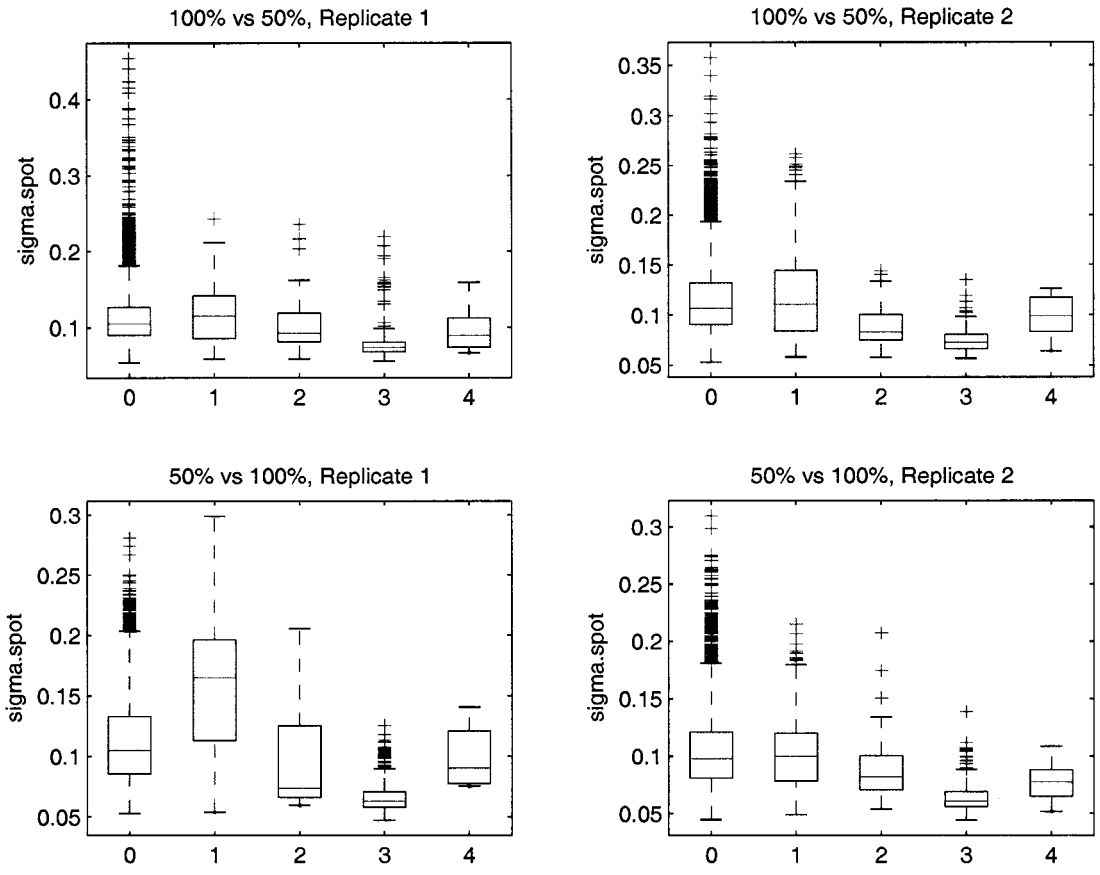


Figure 5.7: For each experiment, the range of σ_{spot} values for each type of SignalViewer flag. The SignalViewer flags are coded as 0=no flag, 1=artifact noise, 2=irregular ellipse, 3=low expression, and 4=no spot detection.

having unreliable pixel intensities.

To further assess the association of SignalViewer and GenePix flags with MSPE and σ_{spot}, ROC curves are used to show predictive power. That is, using an arbitrary threshold c, assign spots a poor quality flag for MSPE $> c$ or $\sigma_{\text{spot}} > c$. The sensitivity and specificity of predicting SignalViewer flags is shown for all values of c and for all four experiments in Figure 5.11. The corresponding ROC curves to predict GenePix flags are shown in Figure 5.12. In keeping with the observations of the boxplots comparing quality scores and flags, similar results are seen here. The MSPE score has some power to predict SignalViewer flags while the σ_{spot} score does not. Neither quality score has an ability to predict GenePix flags,

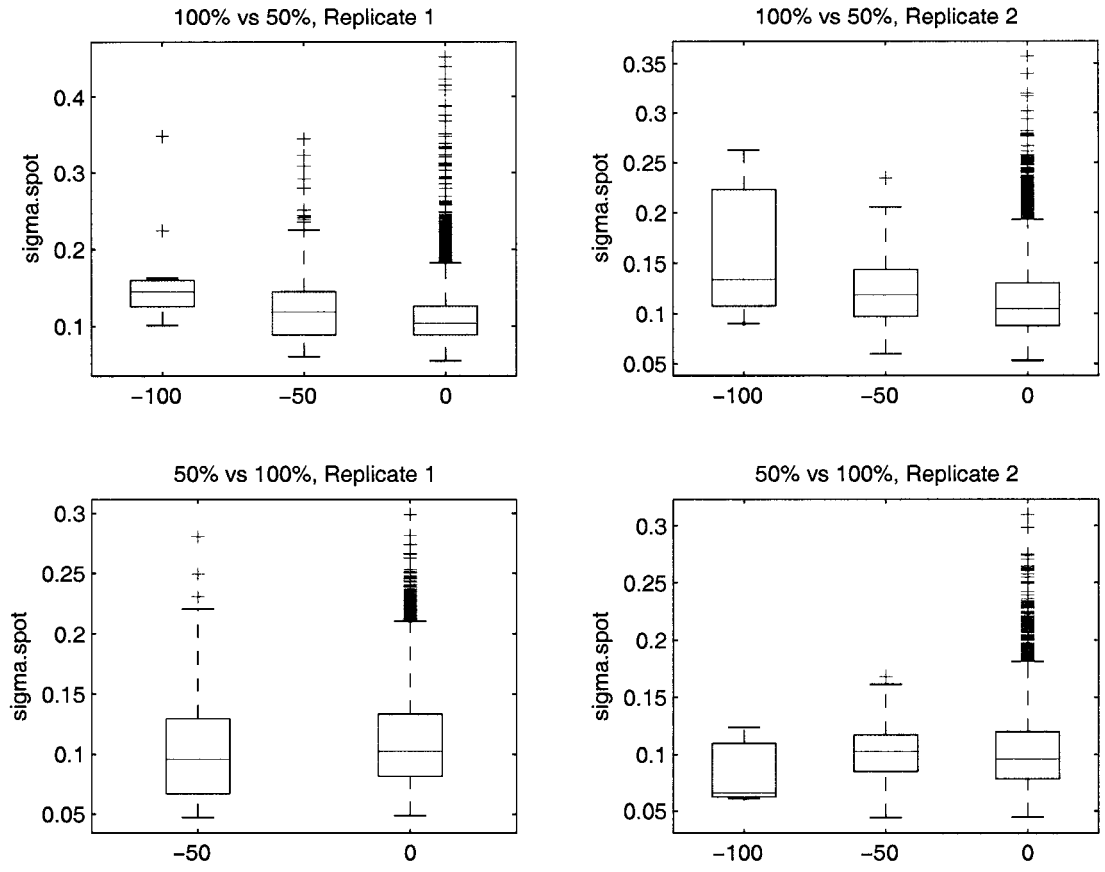


Figure 5.8: For each experiment, the range of σ_{spot} values for each type of GenePix flag. The flags for GenePix are coded as 0=no flag, -50=no spot detection, and -100=manual flag.

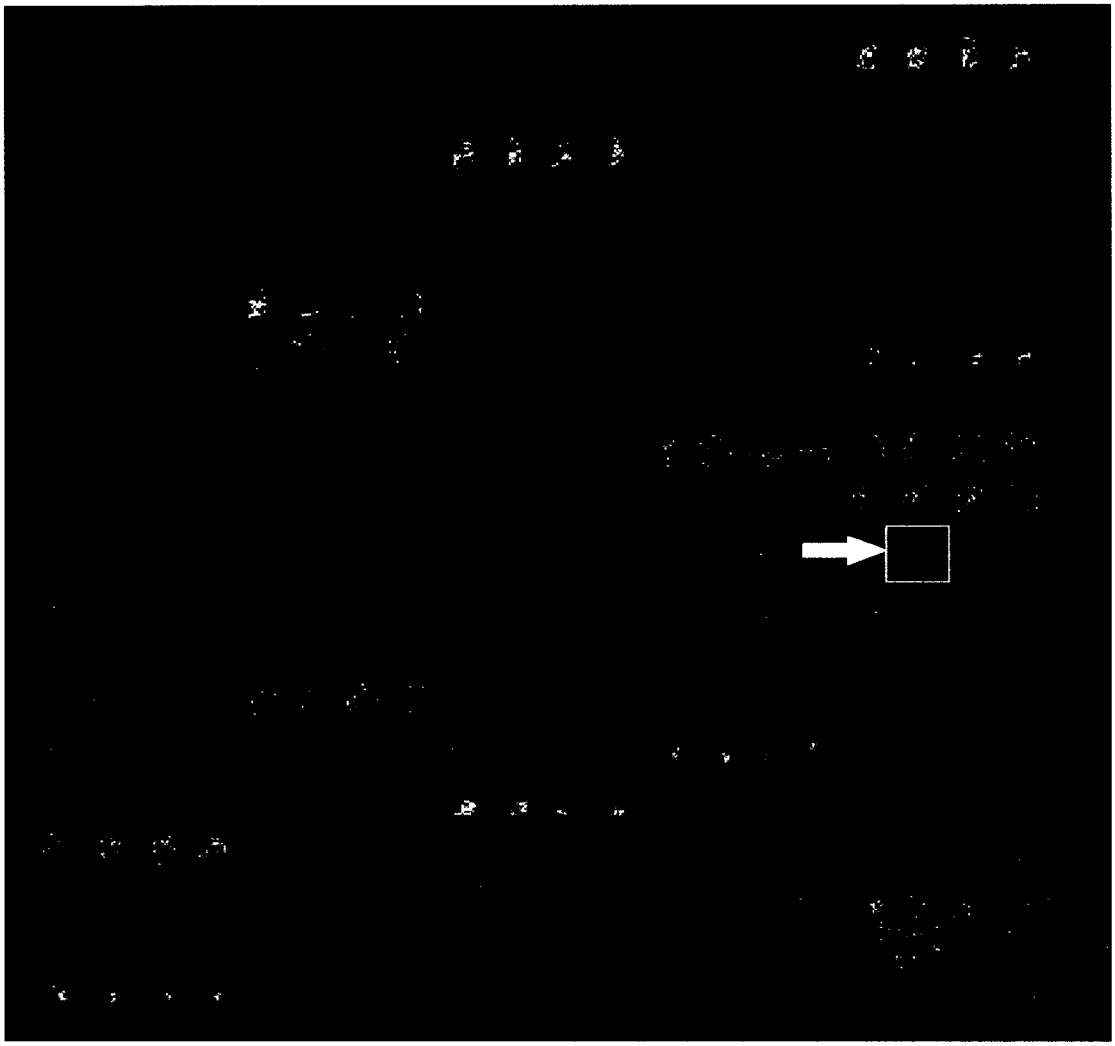


Figure 5.9: A sample block from the second replicate of the 50% to 100% concentration experiment. The green boxes indicate spots assigned a flag of -50 by GenePix. The white box, also indicated with an arrow, shows the spot manually flagged by the FHCRC arraying facility.

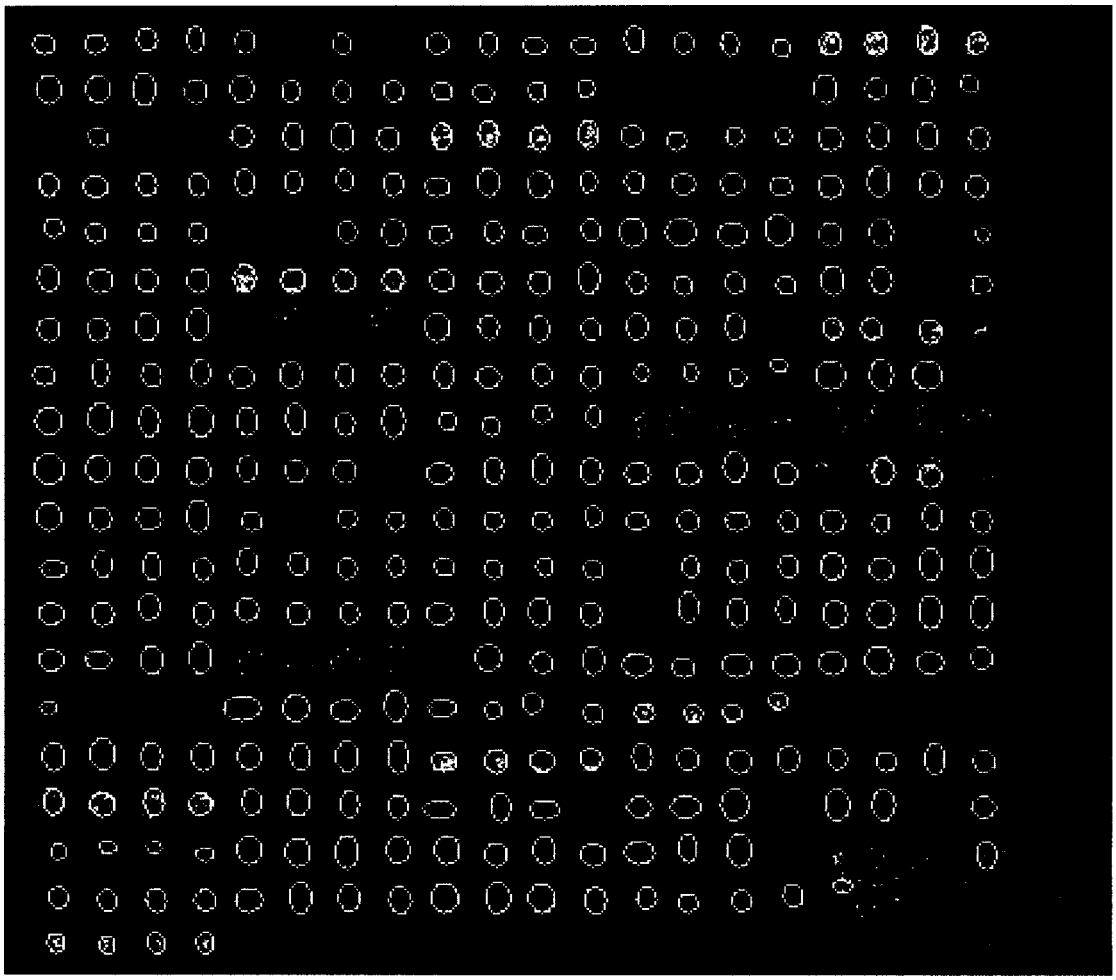


Figure 5.10: A sample block from the second replicate of the 50% to 100% concentration experiment. Red circles indicate spots that SignalViewer flagged. White circles are the results of SignalViewer spot detection.

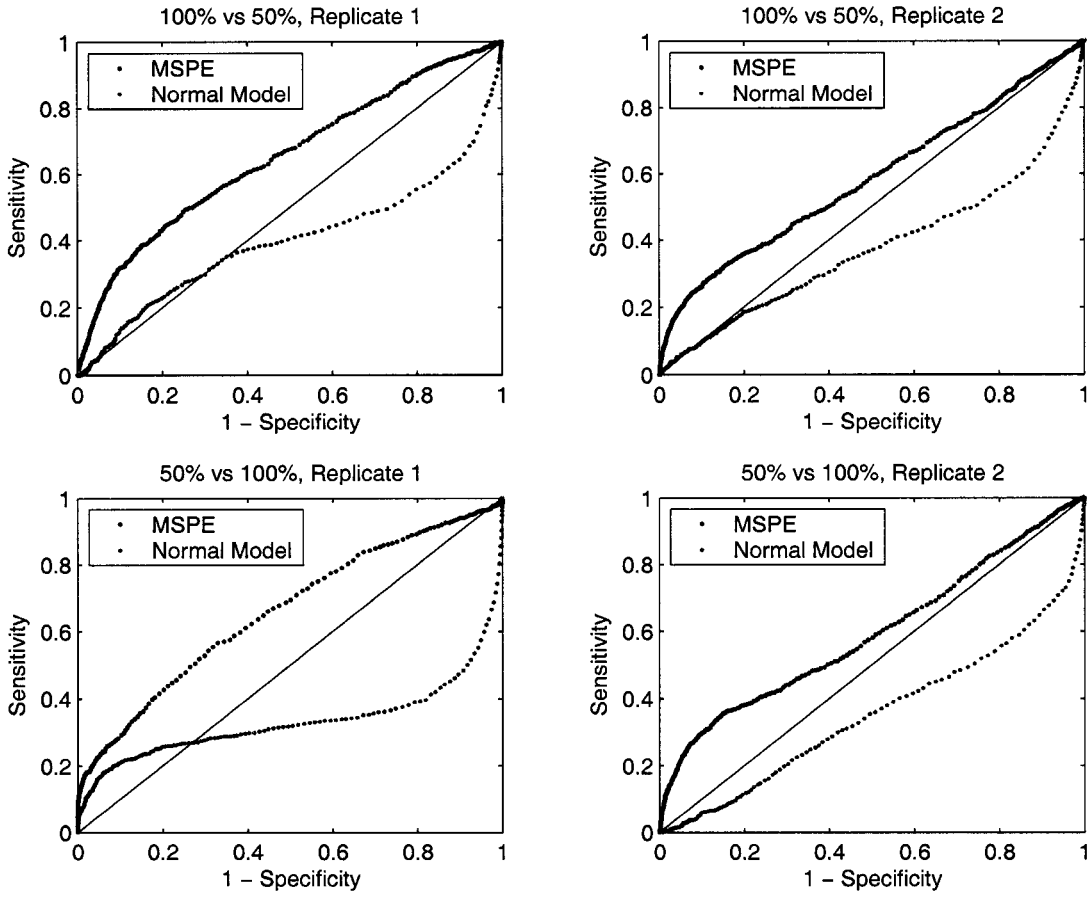


Figure 5.11: ROC curves assessing prediction of SignalViewer flags.

but given their probable lack of reliability discussed above, this is not concerning.

There could be several reasons for the poor performance of σ_{spot} in predicting SignalViewer flags. It could be that the parametric constraints are too restrictive and not representative of actual data. But it could also be that the over-dispersion factor in σ_{spot} that penalizes spots with high levels of spatial correlation does not correspond to the quality expectations in the flagging routines. Whereas the MSPE score, that decreases with increasing correlation, does match up with more ad hoc and intuitive assessments of spot quality.

The end of this section makes a sidenote about the consistency of flagging within Dataset

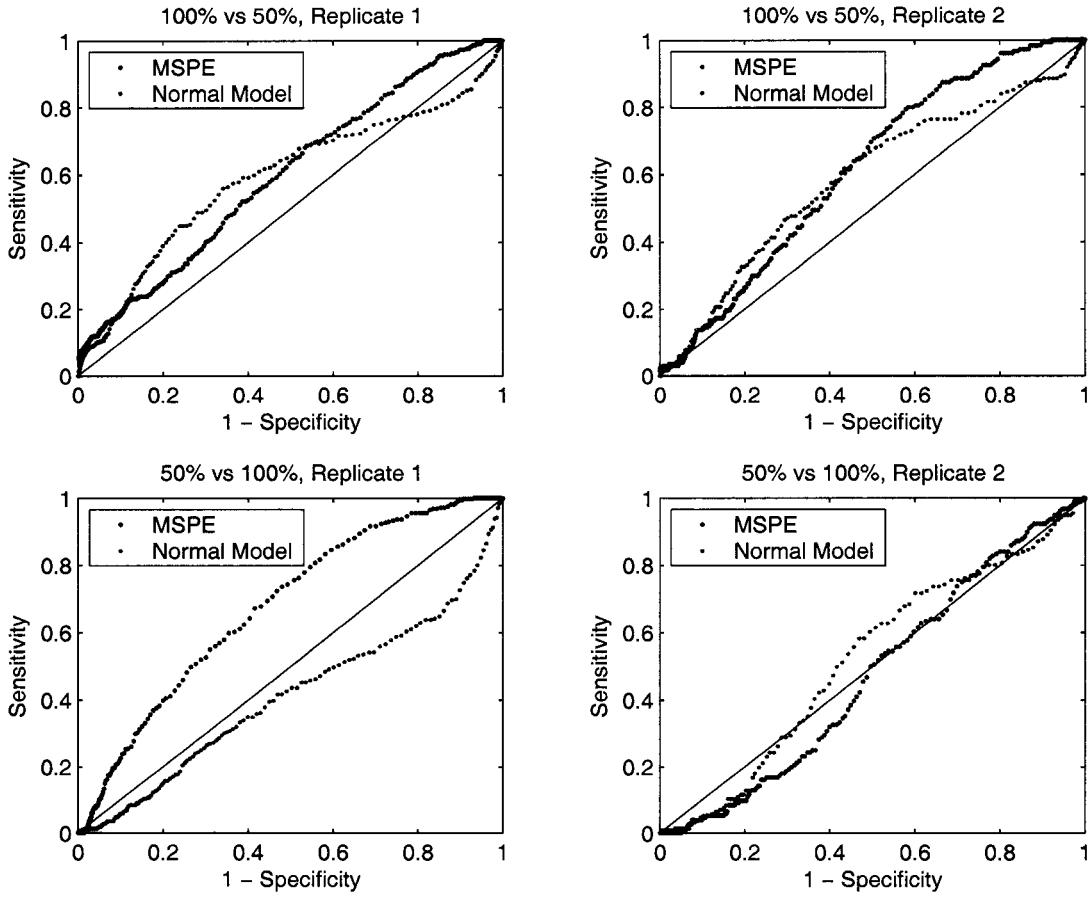


Figure 5.12: ROC curves assessing prediction of GenePix flags.

One. The idea here is that if one spot within a replicate group of four spots is flagged, then its partners will also be flagged. Figure 5.13 shows how often this occurs for four of the experiments in Dataset One when flagging spots using SignalViewer. From this figure, it can be seen that the majority of spots do not have flags. If spots are flagged within a group of four replicates, it is most likely that only one spot has a flag followed by the next most probable occurrence that all four spots in the group have flags. This indicates some consistency for flagging within replicate groups. Naturally, because of variability in printing mechanisms even within replicate spots, flags will not always appear in sets of four.

Figure 5.14 shows the same histograms as in Figure 5.13 where SignalViewer flags have been replaced with GenePix flags. The most obvious conclusion to draw from this figure, as remarked upon before, is that there are far fewer GenePix flags than SignalViewer flags in Dataset One. On average only 4% of spots are flagged by GenePix and 15% of spots are flagged by SignalViewer. The second thing to notice is that consistency in flagging within replicate groups does not hold up as well for GenePix flags.

5.2.2 Spot replication

Taking advantage of the various replications in Dataset One, the standard deviation of replicate spot signals can be compared to spot quality measures. In theory, when spots are of high quality, there will be strong concordance between signals. Thus, high spot quality will correspond with low replicate standard deviation. A low MSPE value for a group of replicates will translate into a low standard deviation. The converse is not necessarily true. A high MSPE value might not correspond to a large replicate standard deviation, but usually does. Similarly, a high replicate standard deviation does not always imply a high MSPE score.

In practice, in Dataset One, this idea holds up. A series of plots, Figures 5.15 through 5.18 support the relationship between both spot quality measures and replicate variability. Replication is examined in two manifestations. First, standard deviations are calculated

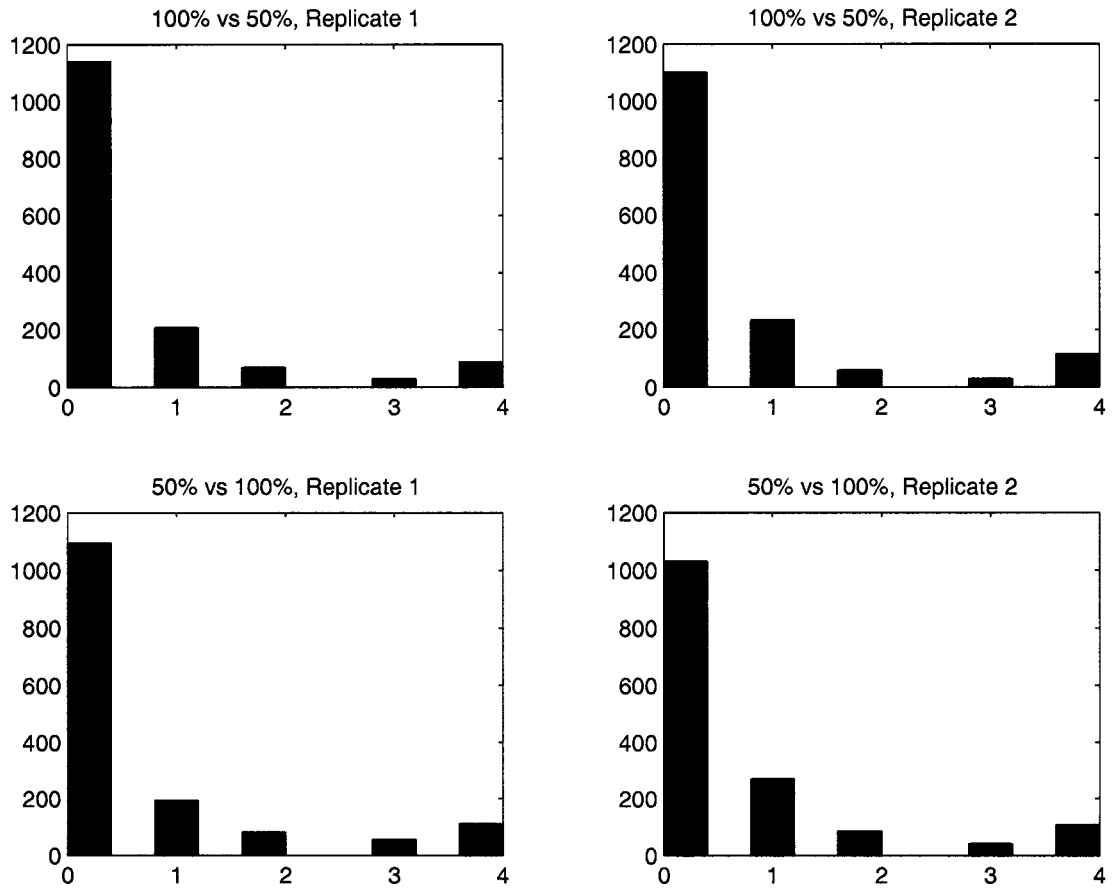


Figure 5.13: For each experiment, the number of SignalViewer flags within a replicate group of four spots. The histogram shows the number of replicate groups that have zero flags, one flag, and so on up to four flags.

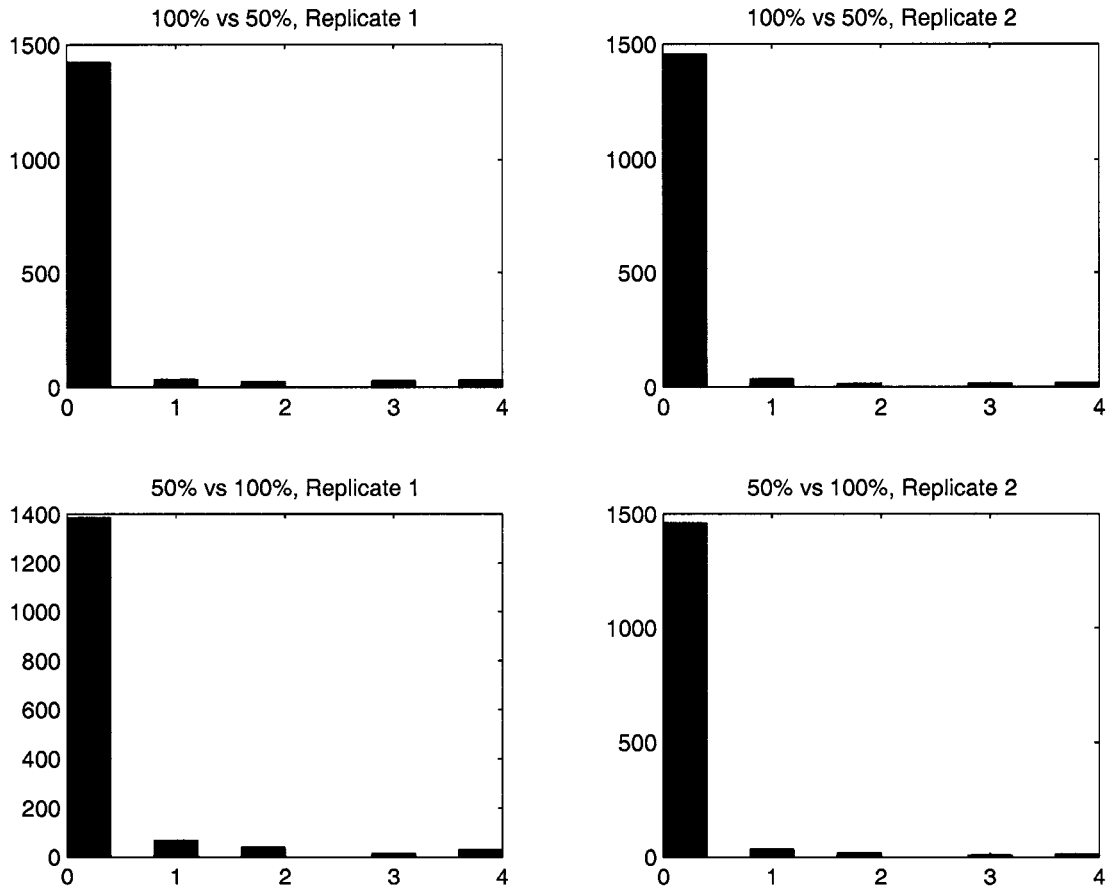


Figure 5.14: For each experiment, the number of GenePix flags within a replicate group of four spots. The histogram shows the number of replicate groups that have zero flags, one flag, and so on up to four flags.

for each group of four replicate spots and plotted against the average spot quality measure for the same group. Figures 5.15 and 5.16 demonstrate this relationship for the four dilution experiments. Both of these figures show that low spot quality values correspond to low standard deviation, as expected. There is also a loose relationship between larger spot quality values and high standard deviation values. Second, standard deviations are computed for a group of 16 replicates. This speaks to the replication over blocks within each microarray. So, a specific gene in the first block is compared to the other 15 replicates spotted in same place for each successive block. The standard deviations for these groups of 16 are plotted against average spot quality in Figures 5.17 and 5.18. Again, the same relationship is observed as with groups of four spots.

The comparison for replicate one of experiment 50% to 100% within Figure 5.18 shows very little association between σ_{spot} and replicate standard deviation. This is the exception to the general pattern. Reasons for this exception would be pure speculation. The majority of the plots, however, show a relationship between spot quality and replicate standard deviation.

The plots in Figures 5.15 through 5.16 also show that the range of spot quality scores, both for the MSPE and σ_{spot}, matches the range of replicate standard deviation values. This indicates that quality plays at least as important a role as replicate variation. If quality scores were much smaller than standard deviation values, they could be disregarded as unimportant in microarray analysis, but this is not shown here.

To examine spot quality in conjunction with replication over entire microarrays, a plot shown in Wang et al is repeated here [50]. For two replicate microarray experiments, the overall correlation coefficient is calculated for the spot signals. Then, after setting a threshold c, the correlation coefficient is estimated only for those spot pairs having an average spot quality value greater than c. As the value of c increases, spot quality diminishes and the correlation coefficient should decrease as well. In fact, this occurs for replicate experiments within both dilution comparisons and is shown in Figure 5.19. The decrease

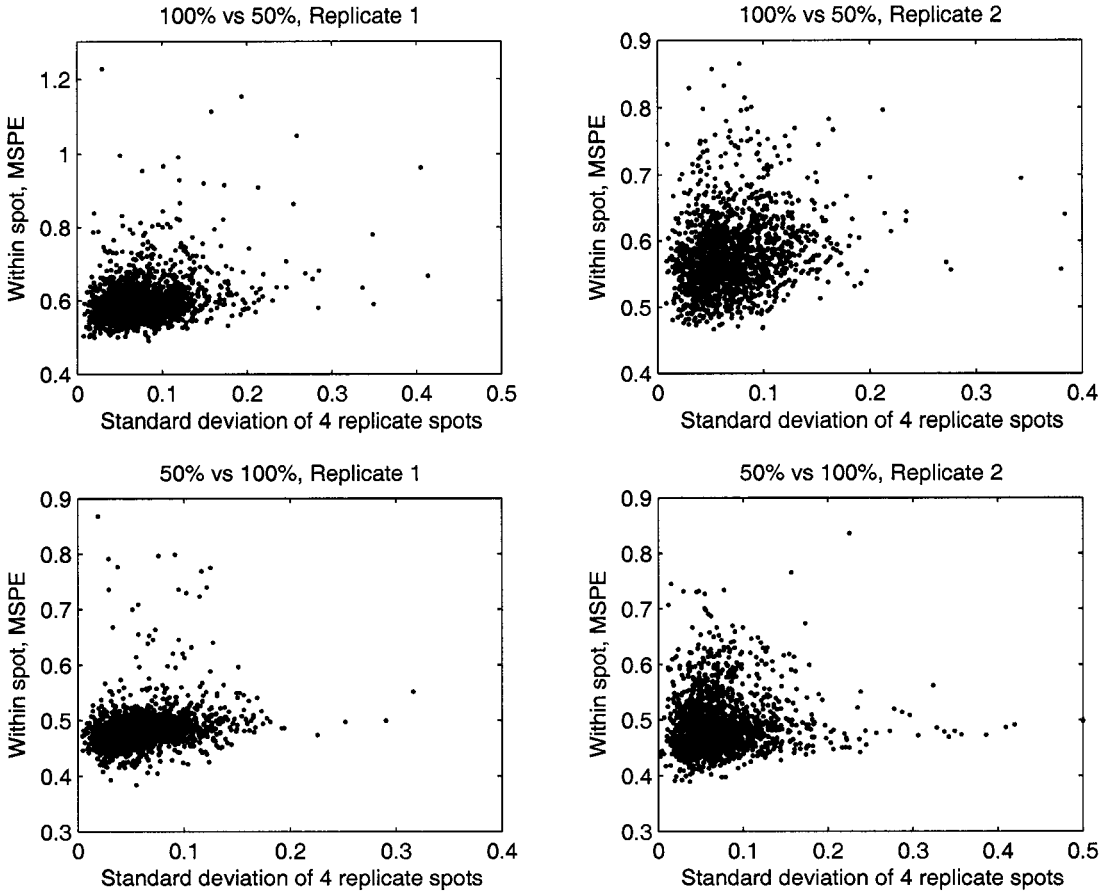


Figure 5.15: For each group of four replicate spots, the average MSPE versus the standard deviation of the spot signal.

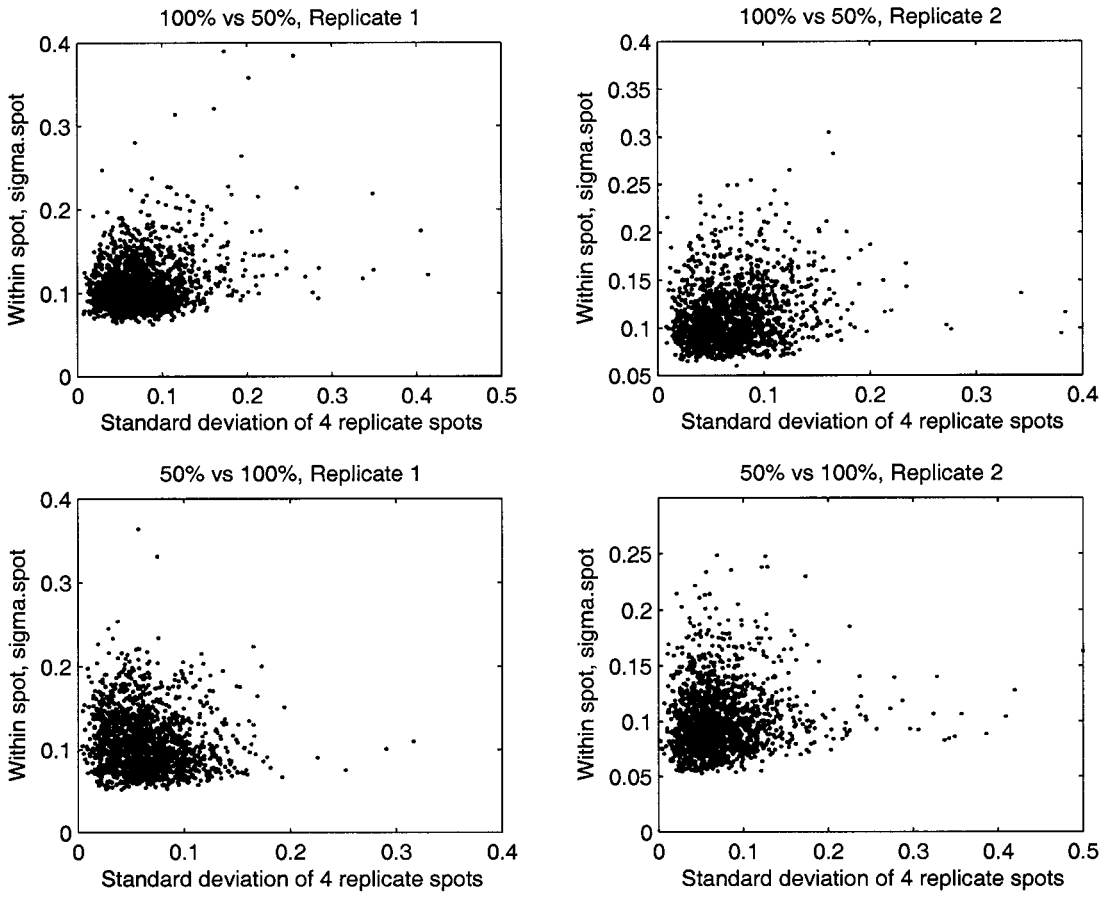


Figure 5.16: For each group of four replicate spots, the average σ_{spot} versus the standard deviation of the spot signal.

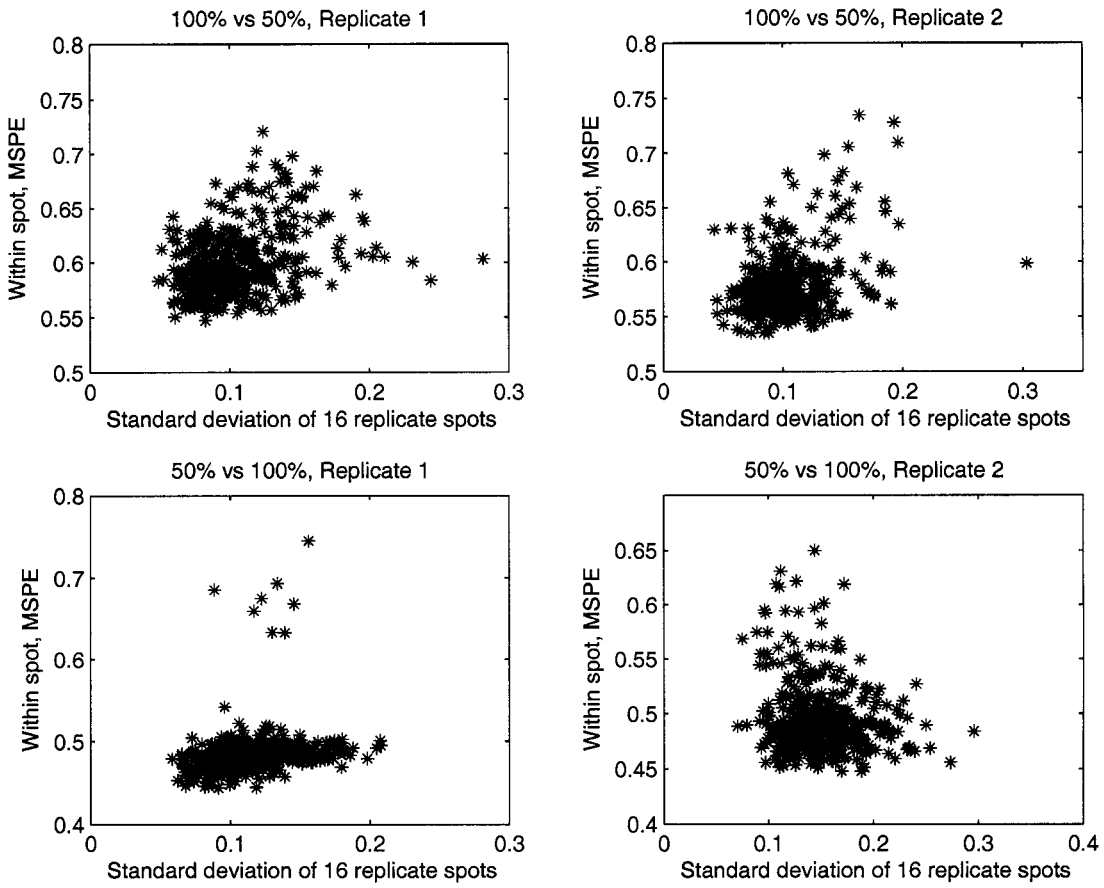


Figure 5.17: For each group of sixteen replicate spots, the average MSPE versus the standard deviation of the spot signal.

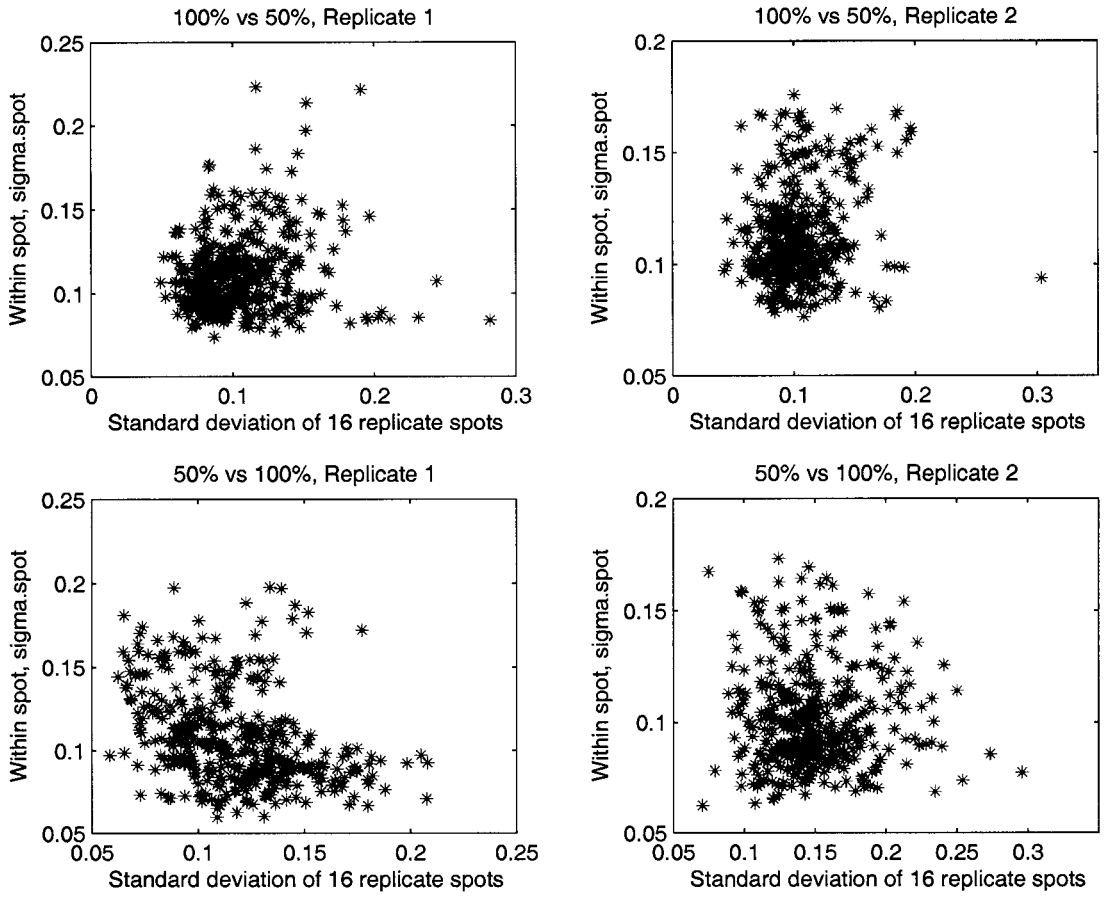


Figure 5.18: For each group of sixteen replicate spots, the average σ_{spot} versus the standard deviation of the spot signal.

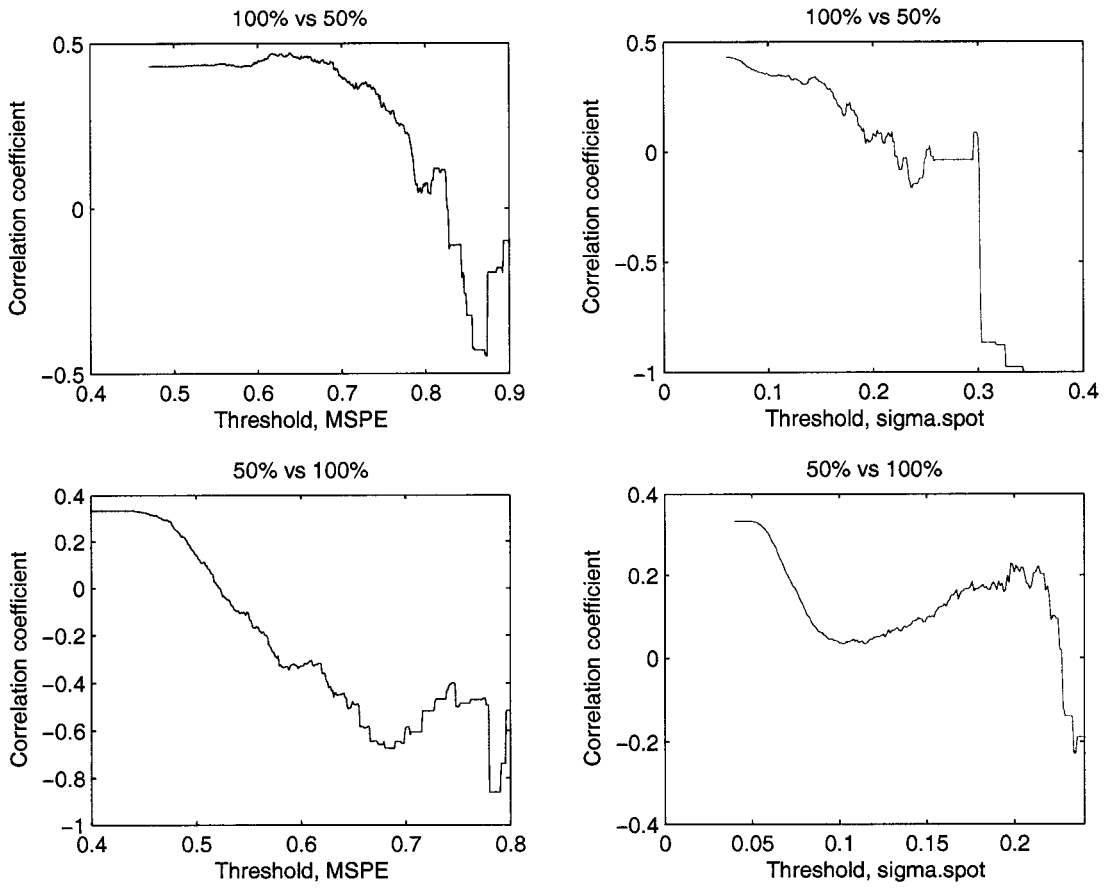


Figure 5.19: Correlation coefficients of spot pairs when their quality measure exceeds a threshold c.

in correlation is more dramatic for the MSPE measure, but for both quality scores, as they increase, the correlation even becomes negative.

5.3 Dataset Two

The point of this section will be to determine the ability of spot quality measures to identify flags for the single microarray image in Dataset Two. Recall that this image was laboriously and manually flagged by lab personnel at the Fred Hutchinson Cancer Research Center using GenePix. Employing ROC curves, these plots will investigate the ability of spot quality scores to predict manual flags (GenePix Flag = -100).

Figure 5.20 shows ROC curves for both the MSPE and σ_{spot}. Clearly, σ_{spot} is unable to detect manual flags. The MSPE score does have some predictive ability, although it is not very powerful. If MSPE scores are supplemented with SignalViewer flags, then the aptitude for determining manual flags increases dramatically. Figure 5.21 illustrates this. Note that three different ROC curves are drawn in this plot. The first curve serves as a reference, using the MSPE measure alone. The other two curves add different kinds of SignalViewer flagging information. The red curve shows the MSPE plus SignalViewer flags for artifact noise, irregular ellipse shape, and spot detection failure. A spot with one of these flags is equivalently given an MSPE value of 1.0. Adding these SignalViewer flags increases sensitivity by about 15% in relevant specificity regions.

The most marked increase in prediction arises when the SignalViewer flag for low expression is added to the MSPE score and other SignalViewer flags. Figure 5.21 shows this with a green curve. The flag for low expression results in a 15% increase in sensitivity alone, compared with the MSPE score and all other SignalViewer flags. All SignalViewer flags together increase sensitivity by about 30% above the MSPE score alone. The jump in prediction for flags of low expression indicates that manual flagging in the microarray facility put some emphasis on marker spots with low intensity levels. But spots of low intensity are not necessarily unreliable data. In fact, spot of low intensity can effectively indicate a lack of transcription for a particular gene.

The final ROC curve in this section depicts the predictive capability of the MSPE and σ_{spot} to determine SignalViewer artifact noise flags in Dataset Two. Artifact noise flags certainly suggest spots of poor quality, unlike flags for low expression or irregular spot shape. Figure 5.22 provides these curves. σ_{spot} is unable to detect artifact noise flags while the MSPE score does have some predictive power.

This particular experiment is a yeast wild type to wild type comparison. In theory, log-ratios should be close to zero. Figure 5.23 shows log-ratio values in Dataset Two versus both the MSPE and σ_{spot}. Spots that have "fold changes" larger than 2 are highlighted in

each plot in color. These highlighted spots all have large spot quality scores indicating that these spots are unreliable data. Hence, spot quality measures can potentially identify wild ratio estimates in this dataset.

In unpublished work, Jinbo Chen and Steve Self used the image in Dataset Two and output from GenePix to develop a CART model that predicts manual flags in this dataset. The optimal CART model used five variables to predict poor spots:

1. the ratio of channel medians
2. the sum of channel medians
3. the red channel spot mean divided by the red channel spot standard deviation
4. the number of background pixels
5. the percentage of spot pixels in the green channel greater than two standard deviations from the green background median.

This CART model has 100% specificity and predicted 110 of the 508 manual flags, resulting in a sensitivity of 21.7%. Although the MSPE and SignalViewer flags used together do not have perfect specificity, the sensitivity is much improved over the optimal CART model. Further, because the CART model uses values in the decision tree that are experiment specific and would very likely vary from array to array and lab to lab, it is not as objective as the MSPE and SignalViewer flags. Finally, the MSPE was not "trained" using Dataset Two while the CART model used to same image to train the model and predict the flags.

A final note ends this section concerning the performance of MSPE scores in ROC curves. Ultimately, spot quality scores are continuous measures and not meant to make a binary prediction of a good or bad spot. Perfect correspondence between spot quality and a binary measure is not expected. Rather, these scores are intended to be used as weights in downstream analysis, as will be discussed in Section 5.4. This allows for more flexibility in modeling data quality without eliminating spots completely and creating missing data.

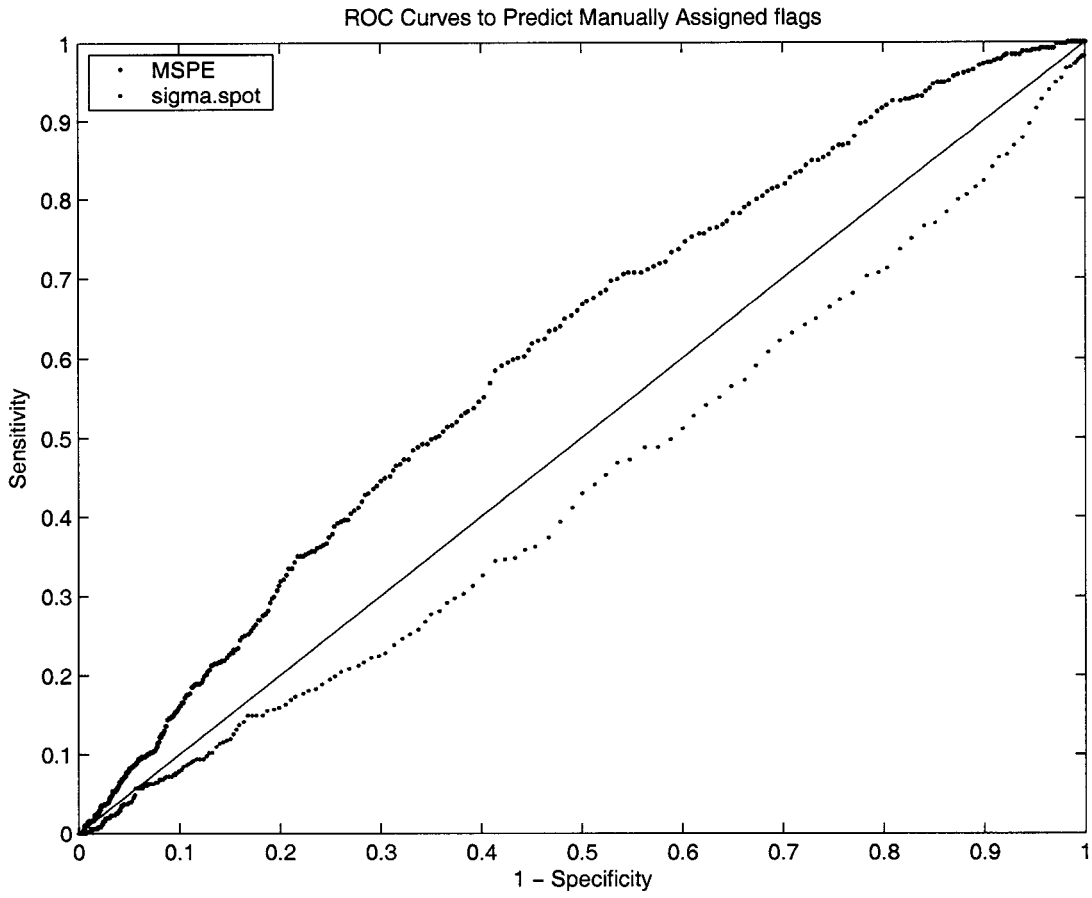


Figure 5.20: Prediction power of MSPE and σ_{spot} to determine manually assigned flags.

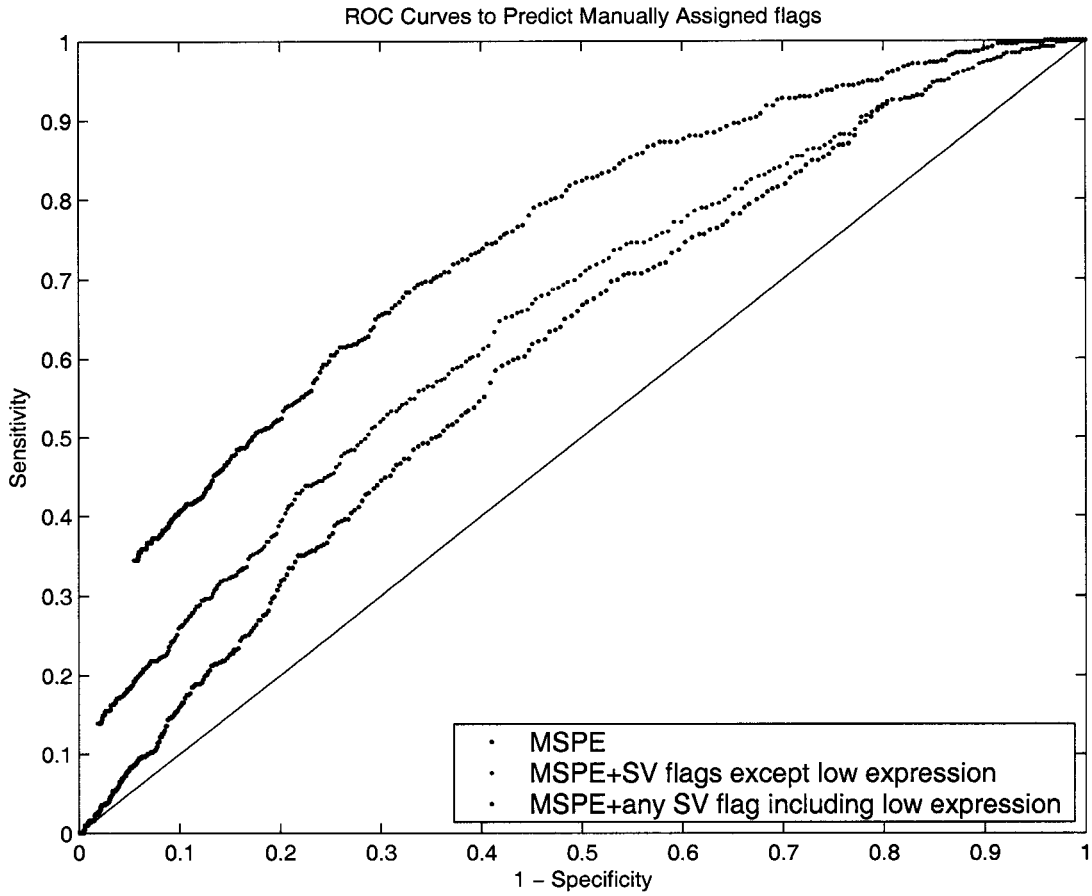


Figure 5.21: Prediction power of MSPE and SignalViewer flags to determine manually assigned flags.

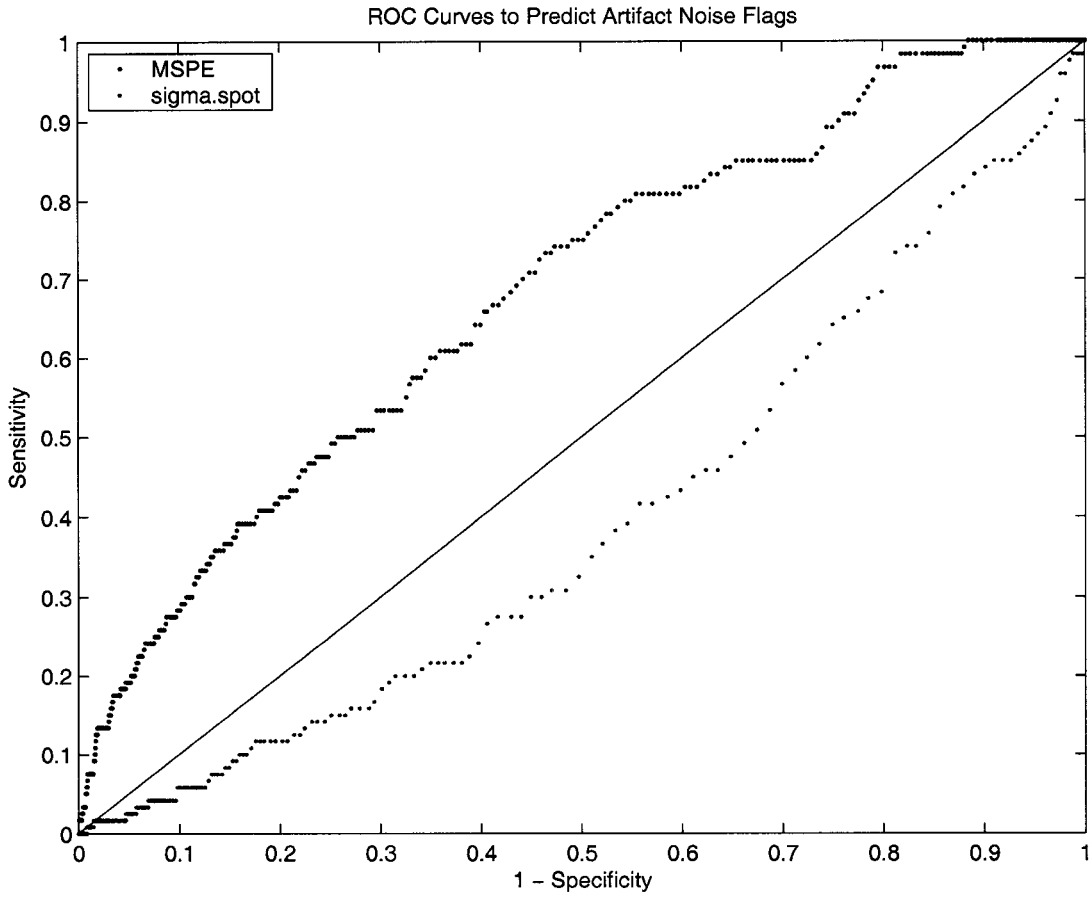


Figure 5.22: Prediction power of MSPE and σ_{spot} to determine SignalViewer artifact noise flags.

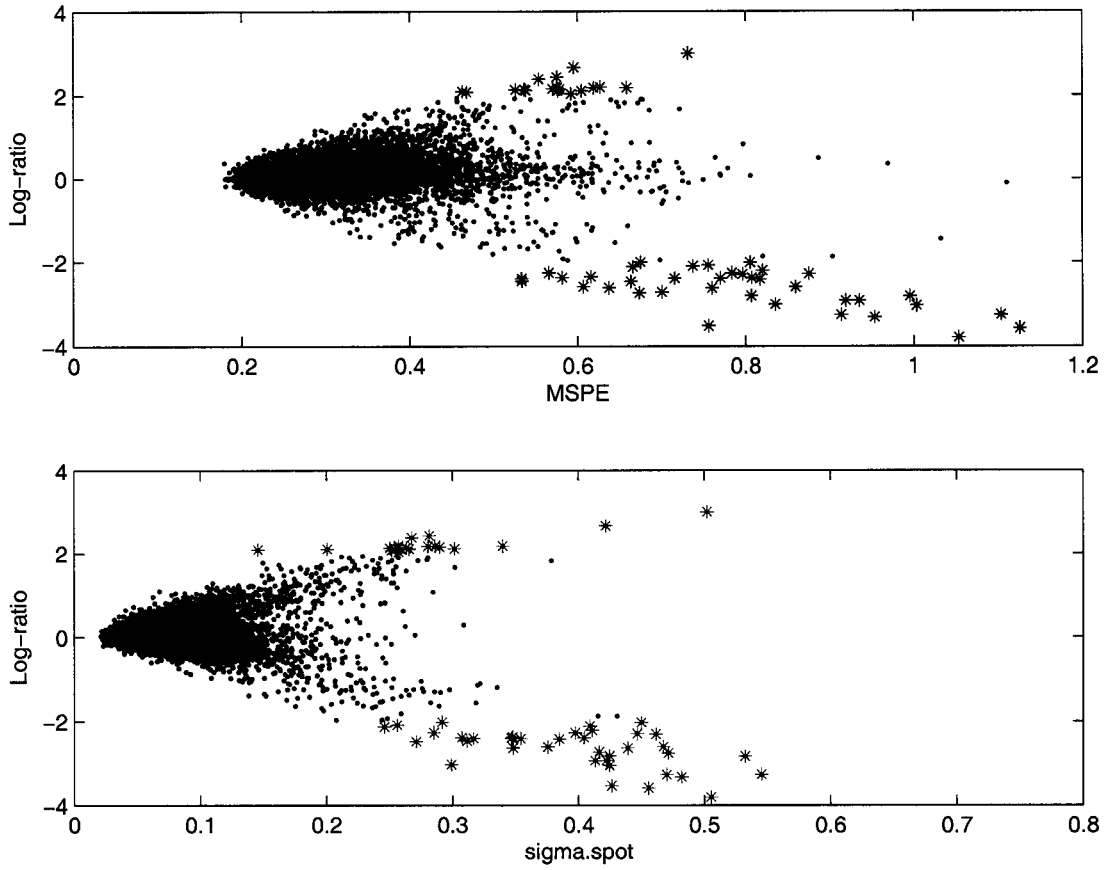


Figure 5.23: Relationship between the log-ratio and spot quality in Dataset Two.

5.4 Downstream analysis

Two main questions arise when pondering the development of spot quality measures: (1) How important is spot quality in comparison to other sources of variation and (2) How can these spot quality measures be used in microarray analysis? Both of these questions will consume the remainder of the discussion in this chapter.

5.4.1 Variance Components

Dataset One replicates spots four times within each block, over each block and over four arrays. This replication makes the data ideal for investigating the importance of different variance components. The size of these components provide information about what aspects of the data need to be accounted for the most. To construct a nested variance components model in an ANOVA setting, let $i = 1, \ldots, 4$ equal the array level, $j = 1, \ldots, 16$ equal the block level, $k = 1, \ldots, 96$ equal the gene level, and $l = 1, \ldots, 4$ equal the spot replication within the gene. Further let Y_{ijkl} represent the log-ratio for each spot within each experiment. The ANOVA model deconstructing each of these components is

$$Y_{ijkl} = \mu + A_i + B_{ij} + G_{ijk} + E_{ijkl}.$$

Table 5.2 gives the sums of squares estimates for each component in the ANOVA model. Nesting is assumed in the experimental design while computing the expected mean squares. Table 5.3 provides the expected values of the mean squares and the solutions for each variance component. The largest variance component is $\hat{\sigma}^2_G = 0.0124$ for the gene effect. That is, the gene factor plays the most important role in explaining the differences in log-ratio values. This is exactly what should happen in microarray experiments. The next most important effects are the array effect, followed by the spot replicate effect and then finally, the block effect.

Using a spot quality measure will capture some of the variability not only in the spot

Table 5.2: Sums of Squares Decomposition for Dataset One

Source	df	Sum of Squares	Mean Square
Total	24575	756.62	
Array	3	174.26	58.09
Block	15	108.74	1.81
Gene	95	344.54	0.06
Spot	24462	129.08	0.007

Table 5.3: Variance Components Estimates for Dataset One

Source	Expected MS	Variance Component	Estimate
Array	$\sigma_E^2 + L\sigma_G^2 + KL\sigma_B^2 + JKL\sigma_A^2$	$\hat{\sigma}_A^2$	0.0092
Block	$\sigma_E^2 + L\sigma_G^2 + KL\sigma_B^2$	$\hat{\sigma}_B^2$	0.0046
Gene	$\sigma_E^2 + L\sigma_G^2$	$\hat{\sigma}_G^2$	0.0124
Spot	σ_E^2	$\hat{\sigma}_E^2$	0.007

effect, but also the gene effect as discussed earlier in this chapter. ANOVA or normalization methods can account for differences at the array level and block level. But clearly, spot and gene effects are important members of the variance components model and adjusting for these effects will impact the resulting analysis.

5.4.2 Use of spot quality measures in downstream analysis

Once a good spot quality score has been derived for all microarray spots, it is of interest to use this score appropriately in downstream analysis. The most intuitive way to use such a score is as a weight. The remainder of this section will focus on microarray analysis that uses a regression setting and how to use a spot quality weight in a regression.

Suppose that $\vec{Y}$ consists of log-ratio values corresponding to microarray spots. Let $\vec{w}$ be a vector containing spot quality scores for each spot, e.g. a vector of MSPE values. If j represents a gene and k is a sample, then Y_{jk} is the log-ratio for a given gene and sample. Further, let $\mathbf{X}_k$ supply covariates for the microarray data including any possible treatment

effects or environmental covariates. The regression model follows.

$$Y_{jk} = \beta_j \mathbf{X}_k + \epsilon_{jk} \text{ where } \epsilon_{jk} \sim (0, \sigma^2 \mathbf{V}_k)$$

The matrix $\mathbf{V}_k$ describes the correlation structure of the gene expression levels. It is within this matrix that spot quality is useful. Here, the diagonal elements are $V_{jk} = w_{jk}$. Often the off-diagonal elements are zero, $V_{jk,j'k} = 0$, but other structures could surely be specified.

To solve parameters in the regression model, the usual weighted regression estimates can be used. In an estimating equations setting, solutions are expanded from a typical weighted regression. This requires solving the score function

$$\vec{U} = \sum_k \mathbf{V}_k^{-1} (\vec{Y}_k - \vec{\mu}_k)$$

and this solution looks like

$$\begin{aligned} \hat{\vec{\beta}} &= (\textstyle\sum_k \mathbf{X}_k' \hat{\mathbf{V}}_k^{-1} \mathbf{X}_k)^{-1} \sum_k \mathbf{X}_k' \hat{\mathbf{V}}_k^{-1} \vec{Y}_k \text{ and} \\ \hat{\text{Cov}}(\hat{\beta}) &= (\textstyle\sum_k \mathbf{X}_k' \hat{\mathbf{V}}_k^{-1} \mathbf{X}_k)^{-1} \sum_k \mathbf{X}_k' \hat{\mathbf{V}}_k^{-1} \vec{r}_k \cdot \vec{r}_k' \hat{\mathbf{V}}_k^{-1} \mathbf{X}_k (\sum_k \mathbf{X}_k' \hat{\mathbf{V}}_k^{-1} \mathbf{X}_k)^{-1} \end{aligned}$$

where $\vec{r} = \vec{Y} - \mathbf{X}\hat{\vec{\beta}}$. Extensions to generalized estimating equations incorporating link functions, where applicable, happen in the usual way.

Using some simulated data, the impact of spot quality scores in microarray analysis can be assessed. Suppose 10 cDNA microarrays are printed with 1000 genes on each array. mRNA extracted from a diseased tissue and a normal tissue are co-hybridized on each array. Of these 1000 genes, suppose that 10 genes are differentially expressed in the two tissue types. Let these 10 significant genes all have good spot quality for each experiment, while all other genes have spot quality values ranging anywhere in the normal range from zero to one. The question is to assess the impact of weights on final Z statistics.

To do this, for each of 10 arrays, the log-ratios of 990 genes were randomly drawn from a $N(0, 3)$ while the 10 significant genes were drawn from a N(2,3). The differentially expressed

genes were randomly assigned quality values, w_{jk} in the range $[0, 0.2]$ and all other genes were given spot quality values in the $[0, 1]$ range. For this simulated dataset, the weighted Z statistics are

$$\begin{aligned}
\hat{\mu}_j &= \left(\sum_{k=1}^{10} \frac{1}{w_{jk}}\right)^{-1} \left(\sum_{k=1}^{10} \frac{Y_{jk}}{w_{jk}}\right) \text{ and} \\
\hat{\sigma}_j^2 &= \sum_{k=1}^{10} (y_{jk} - \hat{\mu}_j)^2 / w_{jk}^2 \text{ and} \\
Z_j &= \frac{\hat{\mu}_j}{\hat{\sigma}_j \sqrt{\left(\sum_{k=1}^{10} \frac{1}{w_{jk}}\right)^{-2}}}.
\end{aligned}$$

Figure 5.24 plots unweighted Z statistics for each of the 1000 genes while Figure 5.25 gives the weighted statistics. The 10 differentially expressed genes are highlighted with red asterisks to contrast with the other genes. These differentially expressed genes definitely have larger Z statistics than most other genes in the unweighted analysis. But there are some other genes with comparable Z scores. The weighted analysis slightly draws the differentially expressed genes away from the rest of the pack and gives them larger Z scores. This speaks to the potential statistical power increase if quality scores are used.

5.4.3 Efficiency of spot quality estimates

This section assumes, as in chapter 4, that spots follow a multivariate normal distribution. Let $\vec{Y} \sim \text{MVN}(\vec{\mu}, \sigma^2\Sigma)$ where

$$\Sigma = \begin{pmatrix} 1 & \rho & \dots & \rho^{D_{ij,kl}} \\ \rho & 1 & \rho & \dots \\ \vdots & \vdots & \ddots & \vdots \\ \rho^{D_{ij,kl}} & \dots & \rho & 1 \end{pmatrix}$$

and $\mu_{ij} = \beta_0 + \beta_1 \cdot I((i,j) \in S)$.

When spots follow this distribution exactly, the true variance of the spot mean, conditional on spot size, is known to be $\text{Var}(\bar{Y}) = \frac{\sigma^2}{n^4}\left[n^2 + \sum_{i,j,k,l=1}^{n} \rho^{\max\{|i-k|,|j-l|\}}\right]$ where (i,j) and $(k.l)$ are coordinates for pixel pairs. The spot has n^2 pixels. The MSPE spot quality

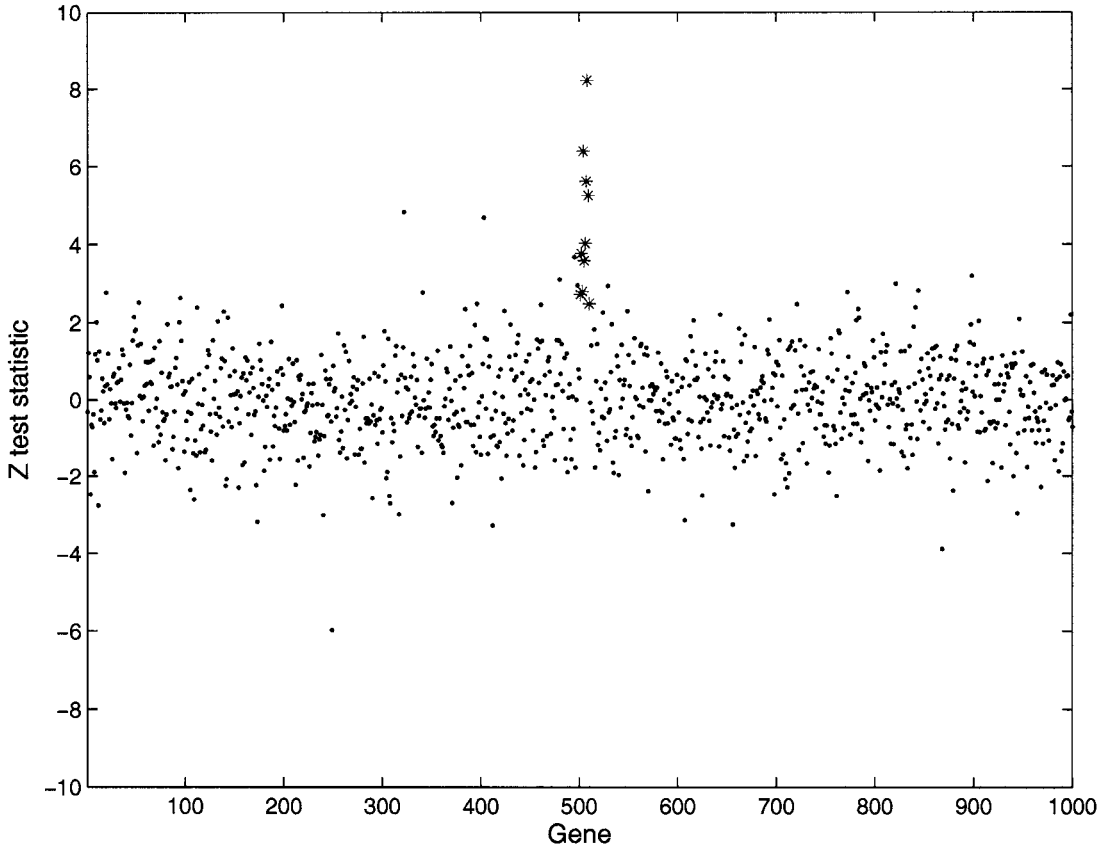


Figure 5.24: Unweighted Z scores for simulated gene expression data. Differentially expressed genes have red asterisks.

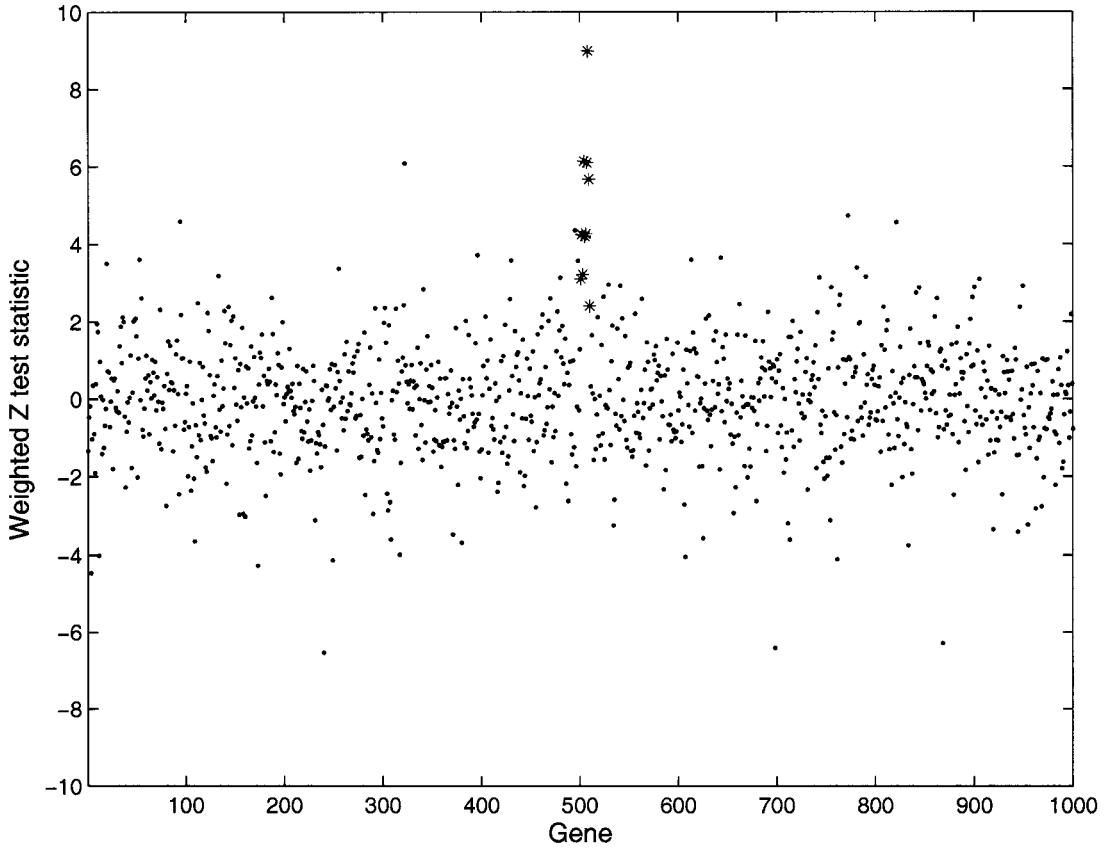


Figure 5.25: Weighted Z scores for simulated gene expression data. Differentially expressed genes have red asterisks.

score, as has been demonstrated in chapter 4, has a different expectation than $\text{Var}(\bar{Y})$. Because the MSPE value is different from the true variance in this setting, the effect of this difference on efficiency in a weighted regression needs to be examined. It is also of general interest to demonstrate improved efficiency over situations where no quality weights are used.

Simulations

The $\text{Var}(\bar{Y})$ is conditional on the sample size, n^2, as well as parameters σ^2 and ρ. By drawing randomly from a empirical distribution for sizes n, spots are generated conditional on this size. The value for $\text{Var}(\bar{Y})$ varies with sample size whereas the MSPE does not. Let the distribution for n be uniform from 10 to 30, encapsulating the normal range of spot sizes.

Suppose that 100 spots are simulated from the distribution described in the previous section. For each simulation, the spot mean and background mean are both equal to one. These 100 spots can be tested for a significant difference from the null where the log-ratio is zero. Let $\sigma^2 = 3$ so that $\vec{Y} \sim \text{MVN}(\vec{1}, 3 \cdot \Sigma)$. For the 100 simulated spots, the denominator of the weighted Z score testing for a difference from zero is computed. That is, the weighted variance of $\hat{\mu}$ is of interest. The first weighted variance estimate uses the true variance as a weight. Therefore $\mathbf{V}$ has components $w_k = \sqrt{\frac{1}{n^4}\left[n^2 + \sum_{i,j,k,l=1}^{n} \rho^{\max\{|i-k|,|j-l|\}}\right]}$. If Y_k is the spot mean and w_k is the corresponding variance or spot quality score, then the weighted variance estimates are

$$\begin{aligned} \hat{\mu} &= \left(\sum_{k=1}^{10} \frac{1}{w_k}\right)^{-1} \left(\sum_{k=1}^{10} \frac{Y_k}{w_k}\right) \text{ and} \\ \hat{\sigma}^2_W &= \left(\sum_{k=1}^{10} \frac{1}{w_k}\right)^{-2} \sum_{k=1}^{10} (y_k - \hat{\mu})^2 / w_k^2. \end{aligned}$$

Let σ^2_V denote a variance using the true expected spot variance as a weight and σ^2_{MSPE} denote the statistic using the calculated MSPE from the simulated spot as a weight. As a side note, if n were constant, the value for σ^2_V is actually equivalent to an unweighted score

because each spot has the same true variance and therefore the same weight. When n is varied, however, then σ_V^2 is different from an unweighted value. One hundred simulation batches of 100 spots each resulted in 100 values for σ_V^2 and σ_{MSPE}^2. Each simulation was run for three different values of $\rho = \{0, 0.5, 0.8\}$, corresponding to no, medium, and high levels of correlation between pixel pairs. To calculate the efficiency of using spot quality scores as weights, the values of σ_V^2 and σ_{MSPE}^2 are compared to an unweighted estimate, σ_I^2.

Results

Table 5.4 shows the resulting variances over 100 simulations for three different correlation levels. The average value of σ_V^2/σ_I^2 and its confidence interval gives the range of the change in efficiency when using the true spot variance. The average value of $\sigma_{\text{MSPE}}^2/\sigma_I^2$ and its confidence interval indicates the change in efficiency when using the MSPE as a weight. When simulated pixels are independent, $\rho = 0$, there is a small gain in efficiency on average when using the true spot variance as a weight. But the simulation confidence intervals cover a wide spectrum in both directions. The efficiency is very close to one when using the MSPE as a weight and the confidence interval is closely knit around the value of one, indicating very little to no punishment when incorporating the MSPE into downstream analysis. As correlation increases to $\rho = 0.5$, efficiency values grow larger than one. That is, efficiency worsens and power will decrease when using the true spot variance as compared to the unweighted scenario. The confidence interval shows that σ_V^2/σ_I^2 is significantly greater than one when $\rho = 0.8$. This adds more evidence to the idea that incorporating spatial correlation as an overdispersion factor penalizes spots with high correlation and leads to decreased power in later analysis. In contrast, efficiency improves when using the MSPE as a weight when correlation is greater than zero. And when correlation is high, for $\rho = 0.8$, the confidence interval for $\sigma_{\text{MSPE}}^2/\sigma_I^2$ is significantly lower than one. This shows that although the expected value of MSPE is quite different than the true variance of the spot, there is little to no penalty, and potentially increased power, for its use in downstream analysis.

Given the empirical observation in chapter 3 that within spot correlation is usually quite large, therefore using the MSPE will tend to result in an efficiency gain.

5.4.4 Conclusions

To investigate the description of data quality using statistical models, parametric, semi-parametric and non-parametric measures of spot variability were explored. Each of these measures were examined for accuracy and stability, performance on real data, and interpretation. Chapter 3 illustrates that microarray spots have insufficient replication to allow for the use of sandwich estimates developed within the semi-parametric framework. Chapters 4 and 5 rigorously investigated the relative advantages and disadvantages of using a parametric approach versus a distribution free approach. The parametric approach employs a well-defined measure of spot variance that conditions on spot size while incorporating a correlation that decays with distance as an overdispersion factor. The distribution free approach employs a proxy to spot variance called prediction error that is more commonly used to assess the performance of smoothing functions. Although the MSPE is not a well understood variance measure, it nonetheless showed a greater correspondence with spot flags, replicate spot variance, and ultimately showed improved efficiency in simulations. Not only is this measure quick to estimate and and stable, but also intuitive to explain to colleagues in the biological sciences while also performing favorably in both Dataset One and Two. Based on these criteria, the author recommends using the MSPE measure to describe spot reliability.

Future work will further investigate the use of MSPE scores on real microarray mea-

Table 5.4: Test statistics for 100 spots. Simulations in batches of 100.

Corr.	Mean Efficiency σ_V^2	95% CI	Mean Efficiency σ_{MSPE}^2	95% CI
$\rho = 0.0$	0.9753	(0.6659,1.2847)	1.0033	(0.9625,1.0441)
$\rho = 0.5$	1.2845	(0.9692,1.5999)	0.9873	(0.9608,1.0137)
$\rho = 0.8$	1.3418	(1.1019,1.5817)	0.9536	(0.9163,0.9909)

surements. It is expected that, particularly when the number of microarray experiments is small and power is an issue, the MSPE used as a weight will significantly increase efficiency. This has yet to be confirmed in actual data sets. It is also of interest to see how this affects downstream conclusions such as the choice of differentially expressed genes between two tissue types.

BIBLIOGRAPHY

[1] MD Adams, SE Celniker, RA Holt, and et al. The genome sequence of *drosophila melanogaster*. *Science*, 287:2185–2195, 2000.

[2] R Adams and L Bischof. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16:641–647, 1994.

[3] P Armitage and T Colton. *Encyclopedia of Biostatistics*. Wiley, New York, NY, USA, 1998.

[4] JB Beckwith and NF Palmer. Histopathology and prognosis of wilms tumors: results from the first national wilms' tumor study. *Cancer*, 41:1937–1948, 1978.

[5] N Brändle, H Bischof, and H Lapp. A generic and robust approach for the analysis of spot array images. *SPIE Proceedings: Microarrays: Optical Technologies and Informatics*, 4266:1–12, 2001.

[6] CS Brown, PC Goodwin, and PK Sorger. Image metrics in the statistical analysis of dna microarray data. *Proceedings of the National Academy of Sciences*, 98:8944–8949, 2001.

[7] J Buhler, T Ideker, and D Haynor. Dapple: Improved techniques for finding spots on dna microarrays. Technical Report UW/CSE/2000-08-05, Department of Computer Science, University of Washington, Seattle, Washington, 2000.

[8] J Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:679–697, 1986.

[9] Y Chen, ER Dougherty, and ML Bittner. Ratio-based decisions and the quantitative analysis of cdna microarray images. *Journal of Biomedical Optics*, 2:364–374, 1997.

[10] VG Cheung, JP Gregg, KJ Gogolin-Ewens, KJ Gogolin-Ewens, J Bandong, CA Stanley, L Baker, MJ Higgins, NJ Nowak, TB Shows, WJ Ewens, SF Nelson, and RS Spielman. Linkage disequilibrium mapping without genotyping. *Nature Genetics*, 18:225–230, 1998.

[11] PA Clarke, R te Poele, R Wooster, and P Workman. Gene expression microarray analysis in cancer biology, pharmacology, and drug development: Progress and potential. *Biochemical Pharmacology*, 62:1311–1336, 2001.

[12] WS Cleveland and Devlin SJ. Locally-weighted regression: An approach to regression analysis by local fitting. *Journal of the American Statistical Association*, 83:596–610, 1988.

[13] KA Cole, DB Krizman, and MR Emmert-Buck. The genetics of cancer - a 3-d model. *Nature Genetics*, 21(supp):38–41, 1999.

[14] The Mouse Genome Sequencing Consortium. Initial sequencing and comparative analysis of the mouse genome. *Nature*, 420:520–562, 2002.

[15] G Delenstarr, H Cattel, C Chen, AN Dorsel, RH Kincaid, K Nguyen, NM Sampas, S Schidel, KW Shannon, A Tu, and PK Wolber. Estimation of the confidence limits of oligonucleotide array-based measurements of differential expression. *SPIE Proceedings: Microarrays: Optical Technologies and Informatics*, 4266:120–131, 2001.

[16] JL DeRisi, VR Iyer, and PO Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278:680–686, 1997.

[17] The C. elegans Sequencing Consortium. Genome sequence of the nematode *c. elegans*: a platform for investigating biology. *Science*, 282:2012–2018, 1998.

[18] B Futcher. Cell cycle synchronization. *Methods in Cellular Science*, 21:79–86, 1999.

[19] A Goffeau, BG Barrell, H Bussey, RW Davis, B Dujon, H Feldmann, F Galibert, JD Hoheisel, C Jacq, M Johnston, EJ Louis, HW Mewes, Y Murakami, P Philippsen, H Tettelin, and SG Oliver. Life with 6000 genes. *Science*, 274:546–567, 1997.

[20] TR Golub, DK Slonim, P Tamayo, C Huard, M Gaasenbeek, JP Mesirov, H Coller, ML Loh, JR Downing, MA Caligiuri, CD Bloomfield, and ES Lander. Molecular classification of cancer: class discovery and class prediction by gene expression monitering. *Science*, 286:531–537, 1999.

[21] AB Goryachev, PF Macgregor, and AM Edwards. Unfolding of microarray data. *Journal of Computational Biology*, 8:443–461, 2001.

[22] DM Green, YA Grigoriev, B Nan, JR Takashima, PA Norkool, GJ D'Angio, and NE Breslow. Congestive heart failure after treatment for wilms' tumor: a report from the national wilms' tumor study group. *Journal of Clinical Oncology*, 21:2447–2448, 2003.

[23] R Haining. *Spatial data analysis in the social and environmental sciences.* Cambridge University Press, Cambridge, UK, 1990.

[24] RG Halgren, MR Fielden, DJ Fong, and TR Zacharewski. Assessment of clone identity and sequence fidelity for 1189 image cdna clones. *Nucleic Acids Research*, 29:582–588, 2001.

[25] S Handran and JY Zhai. Biological relevance of genepix results. *Axon Instruments, Inc.*, page http://, 2003.

[26] T Hastie, R Tibshirani, and J Friedman. *The Elements of Statistical Learning.* Springer-Verlag, New York, NY, USA, 2001.

[27] P Hieter and M Boguski. Functional genomics: It's all how you read it. *Science*, 278:601–602, 1997.

[28] The Arabidopsis Genome Iniative. Analysis of the genome sequence of the flowering plant *arabidopsis thaliana. Nature*, 408:796–815, 2000.

[29] T Kaifel, C Schiekel, and T Kaempke. Spotting approaches for biochip arrays. *IAPR, IEEE Computer Society*, 4:356–361, 2000.

[30] MK Kerr and GA Churchill. Experimental design for gene expression microarrays. *Biostatistics*, 2:183–201, 2001.

[31] MK Kerr, M Martin, and GA Churchill. Analysis of variance for gene expression microarray data. *Journal of Computational Biology*, 7:819–837, 2000.

[32] ES Lander, LM Linton, B Birren, and et al. Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, 2001.

[33] S Lele. Jackknifing linear estimating equations: Asymptotic theory and applications in stochastic processes. *Journal of the Royal Statistics Society B*, 53:253–267, 1991.

[34] KY Liang and SL Zeger. Longitudinal data analysis for discrete and continuous outcomes. *Biometrics*, 42:121–130, 1986.

[35] T Lumley and P Heagerty. Weighted empirical adaptive variance estimators for correlated data regression. *Journal of the Royal Statistics Society B*, 61:459–477, 1999.

[36] DB Martin and PS Nelson. From genomics to proteomics: techniques and applications in cancer research. *Trends in Cell Biology*, 11:S60–S65, 2001.

[37] MJ Martinez, AD Aragon, AL Rodriguez, JM Weber, JA Timlin, MB Sinclair, DM Haaland, and M Werner-Washburne. Identification and removal of contaminating fluorescence from commercial and in-house printed dna microarrays. *Nucleic Acids Research*, 31(4):e18, 2003.

[38] B Merriman. Personal communication. *UCLA*, 2000.

[39] DW Mount. *Bioinformatics.* Cold Spring Harbor Laboratory Press, New York, NY, USA, 2001.

[40] A Narayanan and TW Sager. Table for the asymptotic distribution of univariate mode estimators. *Journal of Statistical Computer Simulations*, 33:37–51, 1989.

[41] WK Newey and KD West. A simple, positive definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica*, 55:703–708, 1987.

[42] JR Pollack, CM Perou, AA Alizadeh, MB Eisen, A Pergamenschikov, CF Williams, SS Jeffrey, D Botstein, and PO Brown. Genome-wide analysis of dna copy-number changes using cdna microarrays. *Nature Genetics*, 23:41–46, 1999.

[43] M Schena, D Shalon, RW Davis, and PO Brown. Quantitative monitering of gene expression patterns with a complementary dna microarray. *Science*, 270:467–470, 1995.

[44] BW Silverman. Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society B*, 47:1–52, 1985.

[45] P Soille. *Morphological Image Analysis.* Springer-Verlag, New York, NY, USA, 1999.

[46] E Southern, K Mir, and M Shchepinov. Molecular interactions on microarrays. *Nature Genetics*, 21(supp):5–9, 1999.

[47] PT Spellman, G Sherlock, MQ Zhang, VR Iyer, K Anders, MB Eisen, PO Brown, D Botstein, and B Futcher. Comprehensive identification of cell cycle-regulated genes of the yeast Saccharomyces cerevisiae by microarray hybridization. *Molecular biology of the cell*, 9:3273–3297, 1998.

[48] JG Thomas, JM Olson, SJ Tapscott, and LP Zhao. An efficient and robust statistical modeling approach to discover differentially expressed genes using genomic expression profiles. *Genome Research*, 11:1227–1236, 2001.

[49] JC Venter, MD Adams, EW Myers, and et al. The sequence of the human genome. *Science*, 291:1304–1351, 2001.

[50] X Wang, S Ghosh, and SW Guo. Quantitative quality control in microarray image processing and data acquisition. *Nucleic Acids Research*, 29:E75–E82, 2001.

[51] X Wang, MJ Hessner, Y Wu, N Pati, and S Ghosh. Quantitative quality control in microarray experiments and the application in data filtering, normalization and false positive rate prediction. *Bioinformatics*, 19:1341–1347, 2003.

[52] G Warnes. The normal kernel coupler: An adaptive markov chain monte carlo method for efficiently sampling from multi-model distributions. Technical Report 395, Department of Statistics, University of Washington, Seattle, Washington, 2001.

[53] H White and I Domowitz. Nonlinear regression with dependent observations. *Econometrica*, 52:143–161, 1984.

[54] YH Yang, MJ Buckley, S Dudoit, and TP Speed. Comparison of methods for image analysis on cdna microarray data. *Journal of Computational and Graphical Statistics*, 11:108–136, 2002.

[55] YH Yang, S Dudoit, P Luu, and TP Speed. Normalization for cdna microarray data. *SPIE Proceedings: Microarrays: Optical Technologies and Informatics*, 4266:141–152, 2001.

[56] LP Zhao, R Prentice, and L Breeden. Statistical modeling of large microarray data sets to identify stimulus-response profiles. *Proceedings of the National Academy of Sciences*, 98:5631–5636, 2001.

VITA

Tracy L. Bergemann is a doctoral candidate in the department of biostatistics at the University of Washington. Originally from Milwaukee, Wisconsin, Tracy did her undergraduate work at Winona State University in Minnesota. During this time, she earned degrees in both mathematics and statistics and a minor in music. Graduate work began immediately thereafter at the University of Washington. Tracy spent three years as a research assistant under the advisement of Norm Breslow working for the National Wilms Tumor Study. This experience provided inspiration to further study the genetics of cancer and to deeply understand the accomplishments in the emerging field of genomics. Then she moved to UCLA in 2000 to study genomics at an Institute for Pure and Applied Math workshop. The past three years have been spent working on dissertation research. In Tracy's spare time she studied the bassoon with Arthur Grossman and played in several bands and orchestras throughout the Seattle area. This past March, Tracy played contrabassoon in the University of Washington Wind Ensemble while on a ten day tour of the Kansai region in Japan.