# COMPARISON OF BACKGROUND CORRECTION IN TILING ARRAYS AND A SPATIAL MODEL

by

DUSTIN MAURER

B.S., Kansas State University, 2006

A REPORT

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Statistics
College of Arts and Sciences

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2011

Approved by;

Co-Major Professor
Haiyan Wang

Approved by;

Co-Major Professor
Susan Brown

# Abstract

DNA hybridization microarray technologies have made it possible to gain an unbiased perspective of whole genome transcriptional activity on such a scale that is increasing more and more rapidly by the day. However, due to biologically irrelevant bias introduced by the experimental process and the machinery involved, correction methods are needed to restore the data to its true biologically meaningful state. Therefore, it is important that the algorithms developed to remove any sort of technical biases are accurate and robust. This report explores the concept of background correction in microarrays by using a real data set of five replicates of whole genome tiling arrays hybridized with genetic material from *Tribolium castaneum*. It reviews the literature surrounding such correction techniques and explores some of the more traditional methods through implementation on the data set. Finally, it introduces an alternative approach, implements it, and compares it to the traditional approaches for the correction of such errors.

# Table of Contents

# List of Figures

# List of Tables

# Chapter 1 - Description of the Data

The ability to sequence whole genomes has been available for some time now. However, the ability to determine which parts of the genomic sequence are important has remained a persistent challenge. In particular, a geneticist may wish to know which sections of the genome are transcribed from DNA, which is held so tightly and securely in the nucleus, into RNA, which floats freely in the cytoplasm. This transcription process is the first among many steps that may eventually terminate in the construction of proteins or other complexes of great importance to the cell. Because not all sections of DNA are transcribed to the same extent, measuring the abundance of the transcripts may also provide valuable information to geneticists.

## 1.1 What is a Microarray?

In an attempt to quantify the transcription of different sections of the genome, microarrays were developed. Microarrays employ a variety of chemical techniques to isolate and measure transcriptional activity including chromatin immunoprecipitation, methyl-DNA immunoprecipitation, DNaseI cleaving, and transcriptome mapping. The microarray technique that is the subject of this study is transcriptome mapping.

## 1.2 Specifics of the Microarrays Under Study

In transcriptome mapping a probe is constructed that represents a length of genomic DNA and is anchored to a position in the microarray. RNA is harvested from the organism, converted to DNA, amplified, marked with photo-florescent molecules, and finally hybridized to the chip. The probes for this study ranged in length from 50 base pairs to 74 base pairs based on their GC content and are spaced about 65 base pairs apart, thus there is occasionally a very small overlap in probe sequence.

Traditionally, oligonucleotide microarrays have been designed with probes of uniform length. This was done in hopes that every probe had a similar bonding strength and for convenience in the manufacture of the probes. However, this procedure leads to a stronger bonding of sequences with a high GC content resulting in a stronger signal by the probe. Probes of similar length with a higher proportion of GC will have a higher melting point than probes with a higher proportion of AT due to the three hydrogen bonds involved in Gs pairing with Cs compared to the two hydrogen bonds involved in As pairing with Ts. By abolishing the constraint of uniform length, probes can be designed that are isothermal. By varying the length of a probe based on its GC content, it is ensured that every probe has a similar melting point and thus the signal of the probe will be a more accurate representation of the presence of the transcript . The probes in this study are isothermal. As support for the isothermal design, a linear model of signal strength dependant on probe length showed probe length to be insignificant in predicting signal strength for the probes in this study.

One additional feature of the microarrays in this study is the inclusion of random probes. This turns out to be a very nice feature that allows for measurement of many different forms of potential error that can occur when trying to measure the presence of transcripts. Many microarrays in the past have neglected to include random probes. This neglect has prevented those arrays from extracting the full potential value from their studies.

The random probes included in these arrays are constructed by randomly choosing among the four nucleotide bases one-by-one and attaching them to the end of a sequence until the sequence has reached the specified melting temperature. In this way probes are constructed that should ideally all have the same binding strength and should not, except in very extremely rare circumstances, be represented in the actual genome. The number of random probes included in the construction of the chips under study is 30,938.

Another useful feature of the microarrays in this study is the random assignment of probes on the chip. There are a total of 4,410,000 (1050 in the X dimension, 4200 in the Y dimension) addresses for probes on the chip. The probes that match unique genomic sequences number 2,142,688 and the number of randomly generated probes, as previously stated, number

30,938.  Each of these probes is randomly assigned to a unique pair of X, Y coordinates on the microarray.  This is done to reduce the effect of any position specific biases that may present themselves.

## 1.3 Sources of Error

Unfortunately, measuring the presence of probe transcripts is not as simple as just reading the signal given by the optical measuring device.  Variability in the signals arise from many sources of error.  These error sources include, but are not limited to, nonspecific binding, optical noise, and position specific bias.

One common feature of micro arrays is optical noise.  Optical noise is a reference to when the light signal read by the machine lens is not exactly the light signal being emitted from the molecules bound to the probe.  Optical noise can come from at least two forms.  One of which is the potential for inexactness within the machine doing the measuring.  Another form of optical noise may be the refraction of light from sources other than the probe under surveillance being taken up by the machine and included in the measured signal for the probe.  Optical noise is sometimes estimated to be small enough and difficult enough to measure to be ignored by many background correction procedures while others other correction measures account for it by subtraction of an array-wide constant.

Nonspecific binding occurs when a sequence that is not exactly complimentary to the probe sequence binds to the probe.  In order to reduce the frequency of this error, heat is applied to the bound substrates.  At temperatures just below the melting point of the full sequence, only those bonds which are strong enough to remain intact, in theory only those which bind to the whole probe, will remain.  This shows the importance of isothermal probes.  Without the property of isothermality, probes with a higher melting point would experience non-specific binding at a higher rate that those with a lower melting point when exposed to the same temperature.

The final source of error is little addressed in microarray literature to date. Position specific bias refers to error that is correlated with a particular absolute address on a chip. Sources of this error are unknown and vary from chip to chip. They could range from uneven heating of the chip, surface tension of the hybridizing solution to the chip, or other artifacts of the experiment or measuring techniques. These position specific biases are the main subject of this study.

## 1.4 Exploration of the Data

Upon receiving a dataset under any context, the first order of business one should undertake is exploration. Tiling arrays are no exception to this rule. An excellent way to form some general ideas about the data is through graphical representation.

I wanted to get a visual of what the original array might have looked like. First I simply plotted the probes as points in a two dimensional square according to their X and Y addresses with the intensity of the probe determining the color of the dot. This method proved far too erratic and any details within the data would have been lost in the mess.

In order to simplify the picture , I decided to separate each dimension into its own graph and plot the medians of all of the probes along each coordinate. What I expected to see was a generally straight line with some variation about a number that would be the median of the whole array. What I actually found took me by surprise. Figure 1.1 shows the medians of all the probes, both random and experimental, with a fitted spline to help identify any trends along each coordinate in both the Y and X dimensions for the third of five technical replicates.

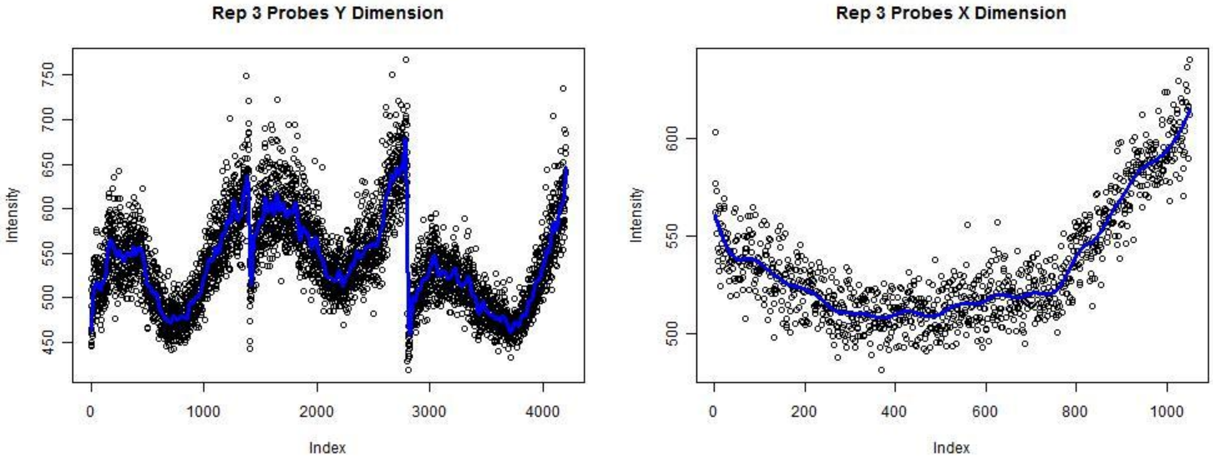


**Figure 1.1 One Dimensional Medians:** The blue line is a fitted spline. There is an obvious curve in these plots of medians.

What became evident in these plots is not only a tendency for some addresses along the X or Y dimension to be greater or less than other addresses along the dimension, but that in the Y dimension there is a clear pattern that repeats from one chip to the other chip for all three chips (the microarray actually consists of three chips laid in a row next to one another).

In order to ensure that the evident patterns are truly something to be concerned about and not some artifact of how the experimental probes were assigned to their respective addresses, I decided to plot only the random probes in a similar fashion (Figure 1.2). The random probes, having been all designed to be isothermal, should have almost exactly the same intensity. Therefore any pattern that exists in the random probes would truly be cause for alarm. While in this plot the shape within the chips of the Y dimension do not show nearly as well, there is still evidence of trends from one end of the Y chips to the other ends of the Y chips as well as the same parabolic shape we saw in the X dimension. I believe that if the experiment contained more random probes, the shapes that were evident in the previous figure would show just as clearly in this figure as well. After all, the sample size is being reduced by about 60 fold.

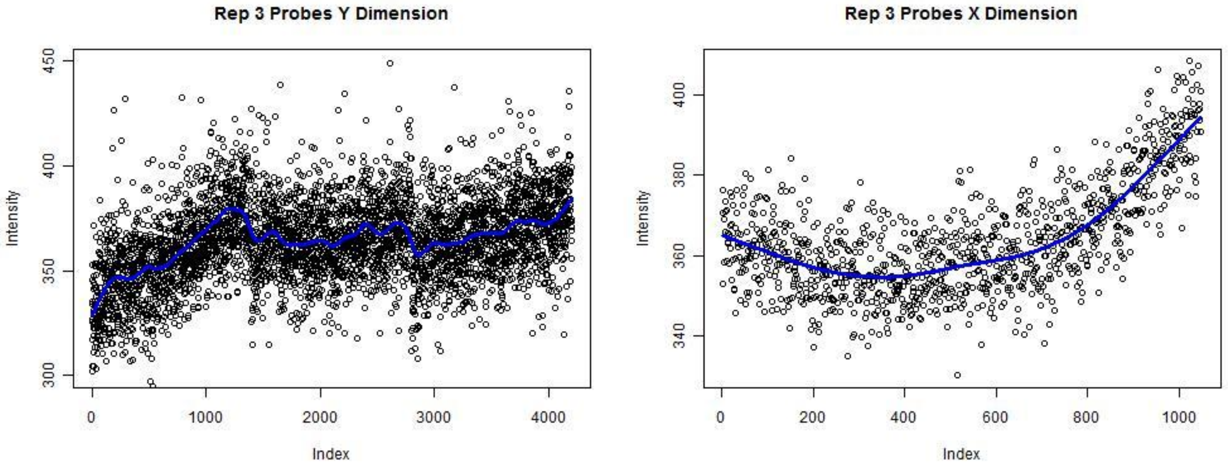


**Figure 1.2 One Dimensional Medians:** (random probes only) The blue line is a fitted spline. A trend is still present, though not as intense as in the plots of all the probes.

Viewing the X and Y dimensions in isolation of one another may provide an alarming picture, however, the microarrays are two dimensional, thus it may make more sense to attempt to visualize both dimensions at once. Figure 1.3 shows a heat map of counts of probes that fall within the first quantile of random probes and reveals that there is a very small number of low intensity probes in some areas, notably the middle right section, while in other areas there is a very high number of low intensity probes, the upper left area.

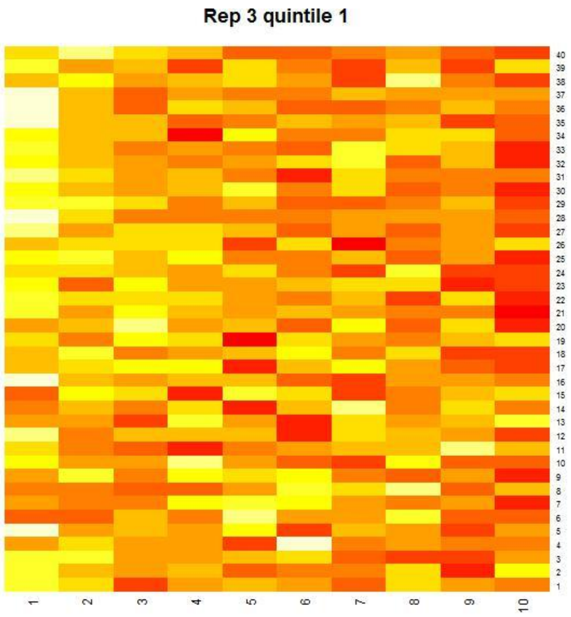


**Figure 1.3 Heat map of Low Intensity Probes:** The first quintile of random probes are represented in 400 evenly segregated bins based on their X, Y address on the chip. Red areas show a lower than average count while white areas show a higher than average count.

Furthermore a three dimensional plot can be developed of the random probe expressions. It is quite evident from the plot in Figure 1.4 that while there is a great amount of variability from probe to probe, there is also an obvious trending upward in intensity as the Y address goes up.

With ample evidence that the nature of the background noise is not consistent across the array, it is worth exploring methods of removing this bias. What follows is a review of the literature surrounding the analysis of microarrays with an emphasis in background correction methods.

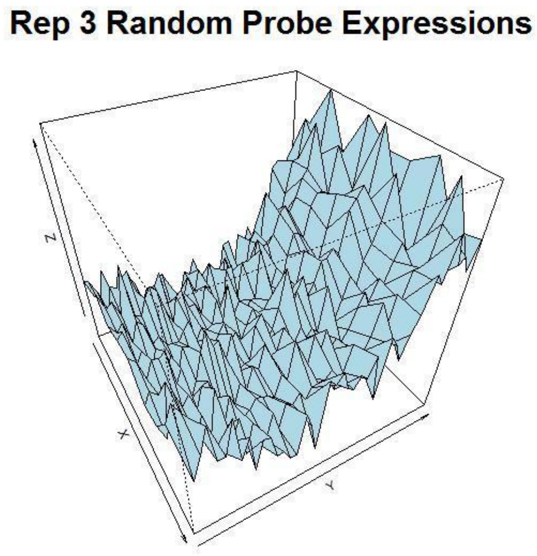


**Figure 1.4 Perspective Plot of Random Probes:** The X and Y axis represent the X and Y address on the array while the Z axis represents the intensity value of the probe.  There is an obvious trend in the Y direction.

# Chapter 2 - Literature Reviews of Background Correction Methods

## 2.1 MAS 5.0

The set of tools developed by Affymetrix for analysis of their own microarray hardware is called the Micro Array Suite 5.0 (MAS 5.0). Recognizing the multiplicative effects of true signal altering factors, MAS 5.0 uses log transforms of signals in their model.

MAS 5.0 relies on the inclusion of mismatch probes (MM) to estimate the amount of background signal for each perfect match (PM) probe. Every probe is of uniform length, 25 base pairs. The MM probes are designed to be identical to the PM probe with the exception of an altered middle (13th) nucleotide.

The MAS 5.0 algorithm depends on two forms of background correction. The first is a location specific correction. For this correction each array is divided into a set of regions, the standard number of regions being sixteen. In each of these regions the lowest 2% of probes are averaged to get an estimation of background noise and the standard deviation of that noise in the region. Each probe is then adjusted based on a weighted average of each of the background values where the weights are determined by the following formula:

$$w_k(x,y) = (d_k(x,y)^{2} + \textit{smooth})^{-1},$$

where k is the centroid, d is the Euclidean distance from the point (x,y) to the centroid, and *smooth* is constant with default value 100.

The weights are determined by the Euclidian distance between the probe and the centroids of the regions. The modified probe intensity value is then determined by the following formula where PM is the resulting intensity value that will be used in all future calculations, P is original probe intensity, *B* is background noise for location (x,y), *D* is standard deviation and *C* is a constant that can be tuned but is set to .5 by default.

$$PM_{x,y} = \max( P_{x,y} - B(x, y), C * D(x,y))$$

The MAS 5.0 method of background correction seems fairly arbitrary and is incredibly dependant on adjusting the tuning parameters to a value such that the desired results are obtained. Moreover, it seems that there is potential that the correction procedure will do nothing more that set a floor for the PM signals.

After the location specific background correction, the second source of background error is sequence related and is estimated with the help of the MM probes. This source of background will be discussed later.

The PM intensity ($I_{PM}$ ) is described as being the sum of two components; stray signal intensity ($I_S$ ) and transcript signal intensity ($I_T$ ). The stray signal intensity is an estimate of the interference of photons from all sources and is unique to each probe.

$$I_{PM} = I_S + I_{T.}$$

The concentration of transcripts of corresponding to MM probes is zero by design, therefore the intensity of true signal corresponding to the MM probes should be zero. Thus, the only factor left in the intensity of the MM probe is the stray signal. It is for this reason that MM signal serves as the initial estimate of stray signal.

Ideally, the MM intensity would simply be subtracted from the PM probe in order to obtain an estimate of transcript intensity for the PM probe. However, this leads to undesirable results in cases where MM is greater than PM. For instance, taking the logarithm of the difference is impossible when it is negative. This leads to a censorship of some of the probes. MAS 5.0 approximates the censorship as uniformly random across all PM probes and assumes that it is independent of the transcript signal for PM probes. This is a reasonable assumption as most cases of MM signal being greater than PM are where there are low concentrations of transcript to be measured by the PM probes.

In order to provide some sort of correction for where the MM intensity is greater than the PM intensity the assumption is made that stray signal is a proportion of observed signal opposed to being a definite value represented by the MM intensity. In cases where the stray signal is considered censored, a new stray signal is computed that is based on the typical stray signal proportion seen for similar PM intensities in the same experiment. After the stray signal is subtracted, what is left of the PM intensity is the true signal.

$$I_T = I_{PM} - I_{S.}$$

According to the MAS 5.0 model, the two aspects of the true signal are the probe affinity (*A*) and the concentration level (*C*). The residual terms (*r*) are assumed to be log-normally distributed. These effects, like every effect in this model, are multiplicative.

$$I_T = C*A*r.$$
$$\log(I_T\ ) = \log(C) + \log(A) + \log(r).$$

Probe affinity is caused by various sources including but not limited to labeling efficiency, binding strength, and density of probe synthesis but the most important factor according to the paper is binding strength. Because the major factor of the affinity term is binding strength it is thought to be nearly constant across experiments.

It is impossible to obtain an absolute affinity term so they are taken to be relative terms that are evenly distributed around zero. Probe affinity is unique to each probe sequence, however, two identical probe sequences found in two different microarrays should have nearly the same affinity factor. A nice property of this constant probe affinity value is that when two identical probes binding to the same transcript have their log-signals subtracted, the affinity term is canceled out.

$$\log(I_T(a)/\ I_T(b)) = \log(C(a)) - \log(C(b))\ \ + r'$$

Here *a* and *b* represent two probes of identical sequence exposed to the same transcript in two different concentrations. The *r'* is the difference between the residuals of probes *a* and *b* and is also normally distributed. Thus the additive model on the log scale allows us to account for the affinity factor when measuring a change in concentration even when the affinity factor is unknown.

In order to estimate both the residual term and the concentration change across experiments, one can subtract the intensities of all corresponding probes of the same gene across two experiments. In this way every probe affinity will have been approximately cancelled out and the concentration differences can be estimated along with the residual term.

The estimator of true signal was established with the idea that it would be stable against multiplicative variation in intensities, robust against outliers, and always positive. The estimate of log-ratio is simply the difference of two experiments using identical probe sets. This also provides a scale on which to judge error and estimate confidence intervals using the t statistic.

The algorithm shows most promise when there is a small amount of corruption in the data. The example given in the paper is a ten percent of the probes have an induced error the algorithm performs better than many mean based estimators. When using MM probes, performance of the algorithm is most improved at the lowest concentration ranges.

MAS 5.0 makes an attempt at position specific background noise signal, however, because there are no probes of random composition included in the arrays for the algorithm to base its correction, the algorithm may be prone to biases introduced by clusters of strong or weak probe intensities. It instead relies on the MM probes to account for the major portion of the background noise for each PM probe. This is perhaps an effective way of accounting for location specific error as MM probes and PM probes are always found adjacent to each other in the design of the arrays. However, in the microarrays of this study, there are no MM probes that can be employed for this purpose.

## 2.2 RMA

Perhaps the most common method used to prepare the raw data of microarrays for further analysis is Robust Multi-array Analysis (RMA). RMA is an expression measure employing a log scale linear additive model. A major motivating factor for using the log scale is the observation that larger mean intensities of probe sets also have larger variances. It was developed with three major criteria in mind of special interest to biomedical researchers, precision, consistency of fold change, and sensitivity and specificity of the method to detect differential expression.

It differs from the MAS 5.0 method which has a model of $PM - MM = \theta\varphi + \varepsilon$ where $\theta$ is the real expression, $\varphi$ is the probe affinity affect, and $\varepsilon$ is an error term. The RMA model is $T(PM) =e+ a + \varepsilon$ where $T$ represents the transformation that background corrects, normalizes, and logs the PM intensities, where $e$ represents the $\log_2$ scale expression, $a$ represents the $\log_2$ scale probe affinity effects, and $\varepsilon$ is an error term. The MM factor was omitted from this model due to empirical results showing that the mathematical subtraction of the MM signal did not translate well to the biological subtraction of the non-specific binding that was supposed to be represented by the MM probe. The omission of the MM sacrifices a small amount of accuracy for large gains in precision.

In the paper (Irizarry et al. *Nucleic Acids Research*, 2003), the RMA algorithm is compared to the dChip and MAS 5.0 algorithms. The RMA procedure outperformed the others in comparing probe signals to their replicates with a statistically significantly higher $R^2$ value than dChip and MAS 5.0. The standard deviation for average expression values was also found to be much smaller and much more consistent under the RMA model than with the dChip or MAS 5.0. The average fold change of genes in different tissues when the microarrays were exposed to different concentrations of cRNA remained more consistent with the RMA procedure than with the dChip and MAS 5.0 methods. Thus the conclusion is reached that RMA is more appropriate method to use for analysis of microarray data than MAS 5.0 or dChip when precision is most important.

Two fundamental aspects of the RMA technique are background correction and quantile normalization. One major assumption that this method of background correction makes is that each array has a common mean background level. The background noise level in RMA would ideally be taken as the mode of the distribution of MM signals (Irizarry et al. *Biostatistics*, 2003). However, in some cases the PM intensity is less than this mode. Therefore, one cannot simply take the log of the PM - mode(MM). In order to overcome this problem the background correction model takes on the form *B(PM)* = E*(s|PM)* where *s* is made to be strictly positive to account for places where the PM is less than the mode of MM. More specifically the E*(s|PM)* term ends up looking like:

$$E(s/PM) = \frac{1}{\Phi(\frac{a}{b}) + \Phi(\frac{s-a}{b}) - 1} \int_0^{PM} \frac{s}{b}\phi(\frac{s-a}{b})ds$$

where $a$ is the parameter estimation of fitting the right tail of the MM probes to the exponential distribution exp( $a$ ), and $b$ is the sample variance of the left tail of the MM probes when fit to the distribution N($\mu$,$\sigma^2$). Here Φ is the standard normal distribution and ϕ is the density function of the standard normal distribution.

Therefore, the true intensity is expressed as PM-*B(PM).* The model applies the largest relative adjustment to the signals of the lowest intensities. This in effect keeps the order of the intensities unchanged.

After background correction and quantile normalization (Bolstad. *Bioinformatics*, 2002), the only factor left in the intensity value that is not part of the true signal is a non-specific binding factor. RMA does not account for this factor explicitly assuming that the genes they are trying to measure are represented by enough probes that the biases in any one probe should be averaged out by the all the other probes in the gene set such that the total bias is near zero.

When making comparisons between MAS 5.0 some very general conclusions can be reached. The inclusion of MM probes in the model of PM intensities can give benefits in accuracy, however, these gains in accuracy come at the cost of precision as well as increased variability in the area of low PM intensities.

## 2.3 GCRMA

The GCRMA model continues in the same vein of other logarithm based models in light of an experiment where the sample hybridized to the array was not related to the probes at all, the resulting signals were approximately normally distributed on the log scale with a very small variance. This experiment shows strong evidence supporting models that take the log transform of the raw signal.

The GCRMA model attempts to improve upon the RMA model. In particular, it hopes to better describe background noise using a statistical model. In Irizarry et. al. (2004) the same experimental data as was used in both the MAS 5.0 and the RMA papers. Background noise is evidenced in these studies by minimum PM intensity that are greater than zero. In addition, probes that had no corresponding transcript to measure displayed a geometric mean intensity of around 200. Initial plots in the GCRMA study show that signal intensity increases linearly with concentration in the original scale but not in the log-scale. This is further evidence of background noise with an additive non-zero mean. This is consistent with additive-background-multiplicative-measurement-error (ABME) models as described in both MAS 5.0 and RMA.

Irizarry makes the claim that the variance of log(PM-MM) (the basis of the MAS 5.0 model) is roughly proportional to the reciprocal of the true amount of RNA. He also finds that log(PM-MM) is a less biased value of the true amount of RNA, but that it generally has a larger variance than simply log(PM), particularly when the true amount of RNA is small.

While RMA outperforms MAS 5.0 and other expression measures in many practical tasks, it only performs a global background adjustment and does not account for non-specific binding resulting in a weakened correlation between nominal and observed concentrations. The GCRMA algorithm is an attempt to improve this correlation.

Irizarry takes some space to examine possible sequence effects on probe binding signals. For instance, G and C will have a stronger bond because of the three hydrogen bonds compared to the two of A and T, however, the U and C bases are the targets that are labeled in amplified RNA studies which in turn impedes binding. Using the data from the above described experiment, they were about to fit a model to determine the effect on total signal of each base in each position. An illuminating graphic shows this relationship.

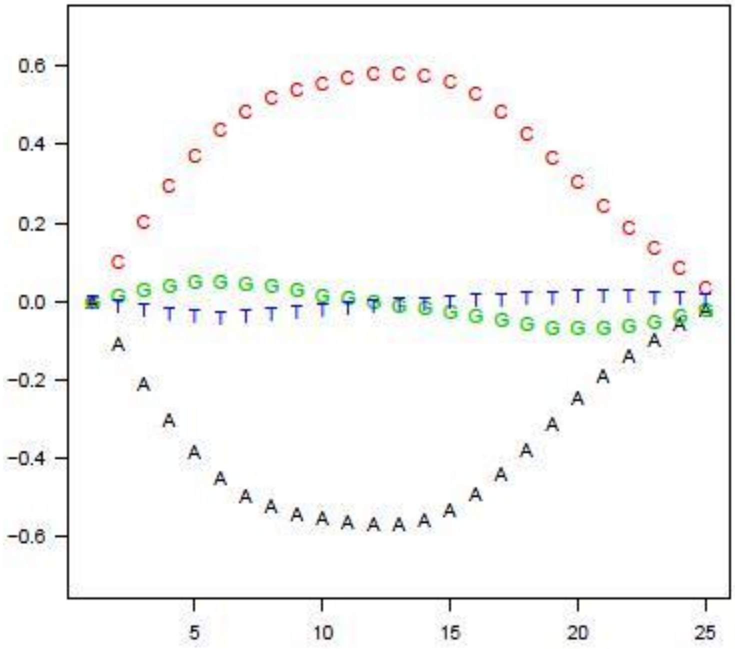


**Figure 2.1 Sequence Effects: The effect of base *A* in position *k*, $\mu_{A,k}$, is plotted against *k*. Similarly for the other three bases.**

Notably, the presence of a C base has the largest positive impact near the center of the probe while the presence of an A base has the largest negative impact at the same positions. The G and T bases have negligible effect throughout.

According to GCRMA the model for the PM probes and the MM probes are as follows:

$$PM = O_{PM} + N_{PM} + S$$
$$MM = O_{MM} + N_{MM} + fS$$

Where O is optical noise which is log-normally distributed, N is non-specific binding noise and is log-normally distributed, and S is the true signal. Here *f* is a value between zero and one and reflects the amount of true signal that also shows up in the mismatch probe signal. In addition, O and N are independent as optical noise and NSB are assumed to be unrelated.

In order to eliminate the background noise, the GCRMA takes on a maximum likelihood approach and an empirical Bayes approach. In the maximum likelihood approach the optical noise and the NSB factor are subtracted out of the PM signal if the difference between the predicted background noise and the actual PM signal are large enough. Otherwise, a value based on the MM signal is subtracted out of the PM signal. This adjusted MM value is shrunken towards the mean of probes with similar affinity levels. The amount of shrinkage is based on the correlation between the PM and MM and is done on a log scale in order to protect against large values of MM resulting from multiplicative error effects.

In all models the optical noise is treated as constant and is estimated as the minimum intensity observed in each array. The non-constant and patterned background noise found in our tiling arrays could be due in part to optical noise, however, the method of simply subtracting out the minimum probe signal from every probe is not a valid solution for our problem as evidenced by the heterogeneous nature of the random probe intensities.

In terms of results, both the maximum likelihood technique and the empirical Bayes technique show similar but slightly better performance than the RMA algorithm in areas such as accuracy of the signal in a single array and accuracy and precision in detecting fold changes, however, the RMA algorithm still shows the most precision in signals on a single array.

## 2.4 Using Spatial Mixed Models

One method employed specifically to model the spatial bias that may be present in microarrays has been the use of linear mixed models. In Baird (2006) just such mixed models are proposed. They opt against smoother models citing an increased robustness in the model at only a small loss of efficiency. They also opt against the use of a cubic smoothing spline because of concerns about PC memory. The model they end up using looks like the following:

$$y = X\tau + Z\mu + \varepsilon$$

where y is the column by column vector of log ratios in array order, $\tau$ is the vector of fixed effects with model matrix X, $\mu$ is the vector of random effects with another model matrix Z, and $\varepsilon$ is a vector of random errors.

The mixed effect model used in this study may have been most appropriate because of the type of array that was being used. In the arrays under study, the set of 8448 probes were being printed onto the array by a 4 x 4 array of print heads giving probes in a 96 x 88 grid. In this fashion, one might expect any source of error from the print heads to have a repeated pattern based on the printing grid. It is for this reason that a linear mixed model was most appropriate in this scenario.

## 2.5 Other Spatial Normalization

More recently, more robust methods of spatial normalization have been developed. Namely, Neuvial and Hupe (2006) produced and algorithm based on using loess trend lines to cluster the probes spatially to determine where potential error may reside as well as correcting these errors or omitting the affected probes from analyses. Their algorithm includes three steps. First, estimate the spatial trend on the surface of the array using two-dimensional LOESS regression. Second, segment the array into spatial areas with similar trend values. Finally, identify and treat the areas affected by spatial bias.

In Fujita (2006), various methods were compared including loess, smoothing splines, wavelet smoothing, kernel regression, and support vector regression. Their conclusions found that loess and smoothing splines performed similarly, however, support vector regression outperformed all other methods.

# Chapter 3 - Proposed Background Correction Method

## 3.1 Verification of Presence of Bias

Running several diagnostics and summary statistics of probe signal strength gave the impression that the data produced by the tiling arrays involved in the study followed the normal trends of many previous microarray studies.  However, the random probes employed in this experimental design allowed for a deeper look into the position specific biases of the probes on the chips.  In particular when mapping the probes to their addresses on the chip and viewing plots of probe strength, certain sections of each chip seemed to exhibit a higher than average signal while certain other sections exhibited a lower than average signal.

The major concern of every previously mentioned background correction procedure was the effect of nonspecific binding.  However, the clever design of setting all probes to the same melting temperature is supposed to eliminate this aspect of background noise.  In addition, given the random assignment of probes to locations on the chip, any noise whose source originated in the design of the probe should exhibit a uniform distribution of the noise across the surface of the chip.  Any noise specific to the design of the probe should only have an effect on the variance of the signal from each of the random probes.  If there was truly a bias in the probe signal based on its location on the chip, the source of the bias would have to be related to the location on the chip.

Previous background correction methods had to rely on experimental probes to give insight into the position specific bias of a chip.  However, due to the fact that previous microarrays rarely, if ever, included a set of random probes, the presence and magnitude of position specific bias would have been difficult to detect and quantify.  With the inclusion of random probes in this study, this task becomes more tractable, and the background noise can be appropriately modeled and removed.

If one were to assume that each array has a common mean background noise level or no background noise, it would be equivalent to stating that differences in background  noise levels will be randomly distributed throughout the chip.  For instance, the top twenty percent of the signals of random probes should be represented in different places around the chip equally and should appear to be randomly distributed throughout the chip, likewise for the bottom twenty percent and for every other quintile as well.  Various methods exist to test the randomness of the distribution of points among a space.  Those included in the second edition of Diggle's *Statistical Analysis of Spatial Point Patterns* include comparing empirical distributions of inter-event distances to simulated data sets that are known to exhibit complete spatial randomness (CSR), comparing nearest neighbor distance to simulations, comparing the distances from a set of points such as constructed by a Dirichlet tessellation algorithm to the nearest event compared to similar points in a simulation, and finally by comparing counts of the number of events in particular quadrants.

The most appropriate of these tests of CSR for this dataset is the comparison of quadrant counts.  Not only does it avoid the tedium and possibly long computing time needed to perform simulations, it can be evaluated very intuitively with the chi-square test which will provide an exact p-value that is easy to interpret and act upon.

In order to perform a chi-square test, quadrant counts need to be established.  The probes must be grouped based on where they lie on the chip.  In addition, a threshold needs to be established for inclusion of probes into their quadrants.  The choice of quadrant size and threshold level are important and related.  Establishing more quadrants will give a more precise description of more slight variations in the position specific bias.  Likewise, establishing smaller layers of threshold will also provide a more precise description of where in the signal level the variation is taking place.  However, given too many quadrants and too small a threshold layer, there may be too many counts with a value of zero, which will reduce the power of the chi-square test and detract from the ability to determine whether or not the data is in fact randomly distributed.  Therefore, it is important to strike a balance between the size of the quadrants and the size of the threshold layer to gain maximum precision and retain maximum power of the chi-square test.

Considering the number of random probes (30,938) and the fact that the microarrays are four times longer in one dimension than in the other (1050 by 4200) the number of quadrants chosen for analysis was 400. This allowed for nearly 80 probes in each quadrant and provided a nice round 10 by 40 pattern for the quadrants. Each quadrant is thus determined by a 105 by 105 square of addresses on the microarray.

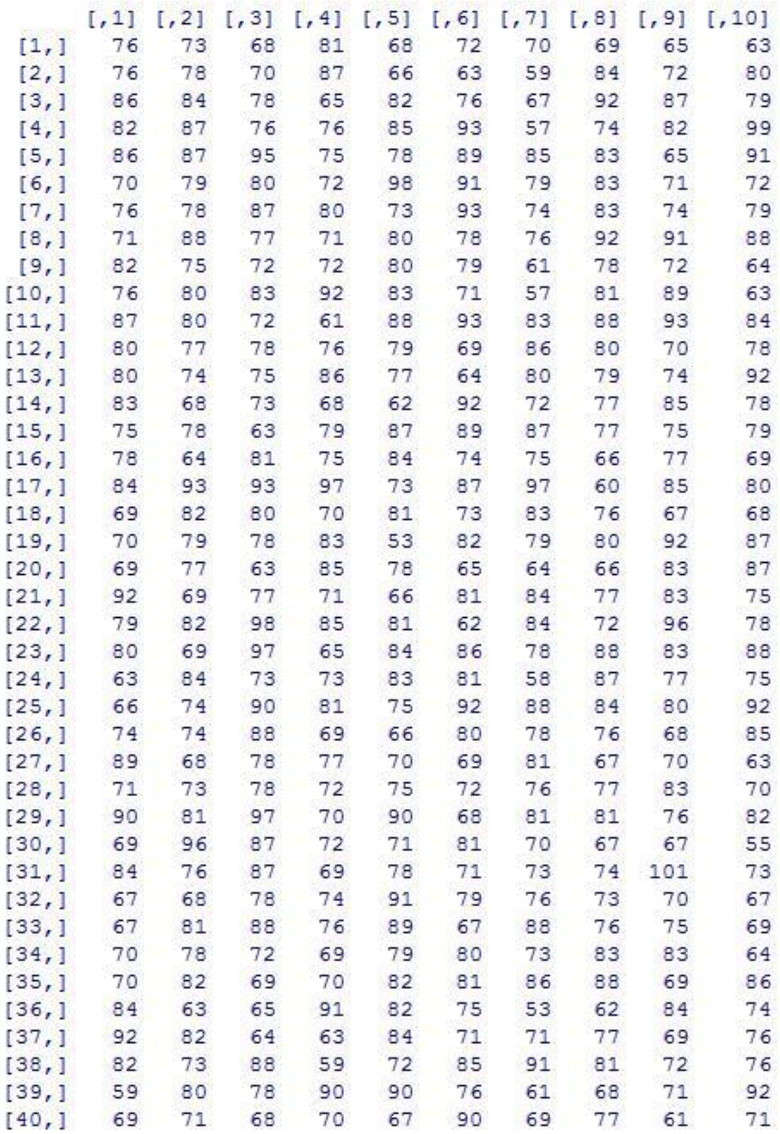
There is no way to reasonably check for quadrant counts of zero other than empirically. Therefore, after testing many threshold levels, the most reasonable threshold levels for these data were established to be twenty percent quintiles. This threshold prevented any of the quadrant counts from being zero. Avoiding counts of zero will ensure that the power of the chi-square test is kept to a maximum.

Once quadrants are established and their counts are determined by threshold levels, the expected value of each quadrant needs to be calculated and the number of degrees of freedom for the chi-square test need to be established. One estimation of the expected count of each quadrant would be the mean of all the quadrant counts. This would be an excellent estimation in a scenario where every quadrant count is expected to be identical, for instance, when testing to see how well all of the random probes are distributed around the microarray in the first place. For this initial test, the degrees of freedom for the chi-square test would be 399 as the only degree of freedom being used by the model is found in calculating the mean of the quadrant counts.

For calculating the expected values of the threshold layers however, there is a more precise method. Since the threshold layer has been determined to be a twenty percent quantile, the expected value of the count of each quadrant for that layer is simply twenty percent of the counts of all probes in the quadrant regardless of signal value. This is a more accurate estimate than the mean of all of the quadrant counts and it does not use any information about the counts of the threshold layer thereby preserving all degrees of freedom. Therefore, for this method we have the same number degrees of freedom as there are quadrants, 400.

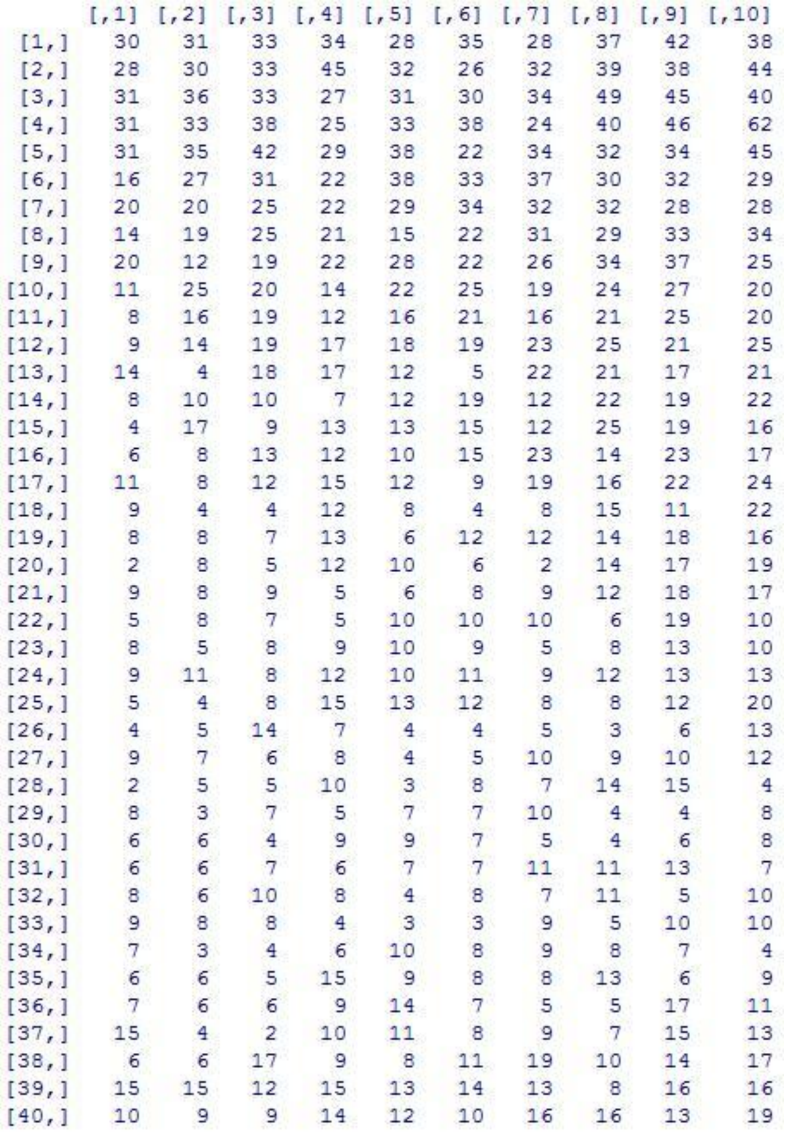
First, the chi-square test was performed on the total quadrant counts. The counts for each of the quadrants in this case is shown in Figure 3.1. Since each chip has the exact same design, these counts are exactly the same from chip to chip. Unfortunately, using the mean of all the quadrant counts as the expected value, the p value for this test is actually less than .05. This is interpreted as the random probes are not very evenly spread throughout the chip. Because of this original misappropriation of random probes, there is potential for the threshold layer counts to be biased as well. This is why the method of calculating expected values based on original quadrant counts was established. By employing this method, the chi-square test remains unbiased by the original design of the chip and distribution of the random probes.

Each of the chips were evaluated at every quintile for CSR using the chi-square test. For example, Figure 3.2 shows the counts of the random probes in the first quintile for the third chip. As is apparent in this figure, there are often places in the chip where bins tended to have fewer counts of random probes at certain quintiles. All chips had very significant p values in the first and last quintiles with less significant p values for the middle quintiles as can be seen from Table 3.1.



```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
 [1,]   76   73   68   81   68   72   70   69   65    63
 [2,]   76   78   70   87   66   63   59   84   72    80
 [3,]   86   84   78   65   82   76   67   92   87    79
 [4,]   82   87   76   76   85   93   57   74   82    99
 [5,]   86   87   95   75   78   89   85   83   65    91
 [6,]   70   79   80   72   98   91   79   83   71    72
 [7,]   76   78   87   80   73   93   74   83   74    79
 [8,]   71   88   77   71   80   78   76   92   91    88
 [9,]   82   75   72   72   80   79   61   78   72    64
[10,]   76   80   83   92   83   71   57   81   89    63
[11,]   87   80   72   61   88   93   83   88   93    84
[12,]   80   77   78   76   79   69   86   80   70    78
[13,]   80   74   75   86   77   64   80   79   74    92
[14,]   83   68   73   68   62   92   72   77   85    78
[15,]   75   78   63   79   87   89   87   77   75    79
[16,]   78   64   81   75   84   74   75   66   77    69
[17,]   84   93   93   97   73   87   97   60   85    80
[18,]   69   82   80   70   81   73   83   76   67    68
[19,]   70   79   78   83   53   82   79   80   92    87
[20,]   69   77   63   85   78   65   64   66   83    87
[21,]   92   69   77   71   66   81   84   77   83    75
[22,]   79   82   98   85   81   62   84   72   96    78
[23,]   80   69   97   65   84   86   78   88   83    88
[24,]   63   84   73   73   83   81   58   87   77    75
[25,]   66   74   90   81   75   92   88   84   80    92
[26,]   74   74   88   69   66   80   78   76   68    85
[27,]   89   68   78   77   70   69   81   67   70    63
[28,]   71   73   78   72   75   72   76   77   83    70
[29,]   90   81   97   70   90   68   81   81   76    82
[30,]   69   96   87   72   71   81   70   67   67    55
[31,]   84   76   87   69   78   71   73   74  101    73
[32,]   67   68   78   74   91   79   76   73   70    67
[33,]   67   81   88   76   89   67   88   76   75    69
[34,]   70   78   72   69   79   80   73   83   83    64
[35,]   70   82   69   70   82   81   86   88   69    86
[36,]   84   63   65   91   82   75   53   62   84    74
[37,]   92   82   64   63   84   71   71   77   69    76
[38,]   82   73   88   59   72   85   91   81   72    76
[39,]   59   80   78   90   90   76   61   68   71    92
[40,]   69   71   68   70   67   90   69   77   61    71
```

**Figure 3.1 The Distribution of the Random Probes in Their Respective Bins:** Each of the numbers represent the count of random probes that are contained within the 105 by 105 coordinate square that is its quadrant.



```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
 [1,]   30   31   33   34   28   35   28   37   42    38
 [2,]   28   30   33   45   32   26   32   39   38    44
 [3,]   31   36   33   27   31   30   34   49   45    40
 [4,]   31   33   38   25   33   38   24   40   46    62
 [5,]   31   35   42   29   38   22   34   32   34    45
 [6,]   16   27   31   22   38   33   37   30   32    29
 [7,]   20   20   25   22   29   34   32   32   28    28
 [8,]   14   19   25   21   15   22   31   29   33    34
 [9,]   20   12   19   22   28   22   26   34   37    25
[10,]   11   25   20   14   22   25   19   24   27    20
[11,]    8   16   19   12   16   21   16   21   25    20
[12,]    9   14   19   17   18   19   23   25   21    25
[13,]   14    4   18   17   12    5   22   21   17    21
[14,]    8   10   10    7   12   19   12   22   19    22
[15,]    4   17    9   13   13   15   12   25   19    16
[16,]    6    8   13   12   10   15   23   14   23    17
[17,]   11    8   12   15   12    9   19   16   22    24
[18,]    9    4    4   12    8    4    8   15   11    22
[19,]    8    8    7   13    6   12   12   14   18    16
[20,]    2    8    5   12   10    6    2   14   17    19
[21,]    9    8    9    5    6    8    9   12   18    17
[22,]    5    8    7    5   10   10   10    6   19    10
[23,]    8    5    8    9   10    9    5    8   13    10
[24,]    9   11    8   12   10   11    9   12   13    13
[25,]    5    4    8   15   13   12    8    8   12    20
[26,]    4    5   14    7    4    4    5    3    6    13
[27,]    9    7    6    8    4    5   10    9   10    12
[28,]    2    5    5   10    3    8    7   14   15     4
[29,]    8    3    7    5    7    7   10    4    4     8
[30,]    6    6    4    9    9    7    5    4    6     8
[31,]    6    6    7    6    7    7   11   11   13     7
[32,]    8    6   10    8    4    8    7   11    5    10
[33,]    9    8    8    4    3    3    9    5   10    10
[34,]    7    3    4    6   10    8    9    8    7     4
[35,]    6    6    5   15    9    8    8   13    6     9
[36,]    7    6    6    9   14    7    5    5   17    11
[37,]   15    4    2   10   11    8    9    7   15    13
[38,]    6    6   17    9    8   11   19   10   14    17
[39,]   15   15   12   15   13   14   13    8   16    16
[40,]   10    9    9   14   12   10   16   16   13    19
```

**Figure 3.2 The Distribution of the Random Probes of the Third Replicate, First Quintile in Their Respective Bins:** Each of the numbers represent the count of random probes that are contained within the 105 by 105 coordinate square that is its quadrant. The numbers in this particular figures are representative of the bottom twenty percent of signals of random probes in the third replicate.

|  | **Microarray 1** | **Microarray 2** | **Microarray 3** | **Microarray 4** | **Microarray 5** |
|---|---|---|---|---|---|
| **1st quintile** | 4.007569e-17 | 1.040798e-39 | 6.987812e-47 | 9.267160e-217 | 7.455458e-36 |
| **2nd quintile** | 1.214135e-02 | 8.596984e-04 | 1.895252e-06 | 1.581560e-43 | 8.628489e-01 |
| **3rd quintile** | 9.860017e-01 | 7.759306e-01 | 9.681052e-01 | 8.810283e-01 | 9.994862e-01 |
| **4th quintile** | 9.643598e-01 | 3.501207e-02 | 8.738846e-02 | 6.244116e-26 | 2.548600e-01 |
| **5th quintile** | 6.093433e-105 | 5.779450e-30 | 2.443781e-114 | 0.000000e+00 | 1.126680e-12 |

**Table 3.1 P-values for the Chi-Square test of CSR:** The first and last quintiles show significant p-values while many of the inner quintiles don't. This is evidence of a general shift upwards or downwards of signal intensity in some areas of the array.

The tendency of the p-values to be near zero in the first and fifth quintile and near one in the third quintile is an indication of general shifts of intensities either up or down and also that there is healthy variability within each quadrant. If each quadrant consisted of intensities clustered closely around a certain value with very little variance, the chi square table may have shown significant p-values in every quintile.

## 3.2 Establishment of The Model

After demonstrating that the signal strength of a probe is not independent of its position on an array, the next step towards eliminating this error is to attempt to model it. By modeling the position specific bias of the probe, the degree to which the signal strength is affected by its location on the chip can be estimated. After an estimation of bias is obtained, it can be subtracted from the observed signal to yield the unbiased signal strength.

There are many different methods by which the position specific bias can be modeled. In order to decide on the appropriate method, concerns about the shape of the bias surface and the end use of the model estimates must be taken into account. To get an idea of the bias surface, a three dimensional graphic may be drawn for each probe. The end use of the modeled estimates will give an idea of how important avoiding over fitting might be.

A visual inspection of the three dimensional surfaces of the random probes, as seen in chapter one, reveals great deal of information. It becomes apparent that not only do the surfaces often fail to conform to any sort of symmetric shape, but often there are patches of random probes that show a much higher signal than their surrounding counterparts. Another interesting observation is that from chip to chip the surfaces bear little or no resemblance to each other. This lack of similarity between chips confirms the notion that each chip will require its own model parameters opposed to creating one model to represent every chip. The position specific bias of each chip should be modeled independently of every other chip.

Some issues may be raised with the idea of modeling each chip independently. For instance, each position with a random probe will only have one observation. Unfortunately, this means that there is no way of obtaining an estimate of variance on the probe signal. However, due to the deterministic nature of the chemistry involved, it can be fairly certain that the variance of the probe signal is quite small, at least small enough that it will not be significant in comparison to the difference in signal strength of two different probes. Therefore, it is safe to take the signal of the probe to be fairly accurate estimation of the position specific bias at its point on the chip.

Thus, the main objective of the model will be to describe the bias present at a particular position on the chip as judged by the signal given by the random probe at that location. Since the model will only be used in describing the bias of a particular chip and will not attempt to describe the biases in all chips, there is small concern for overfitting. The model must be smooth enough not to be swayed by random variation that may occur within the random probe signals, but still be specific enough to accurately describe the shape of the bias. Therefore, it may be best to take a nonparametric approach. This will allow the curve to be specific at a local level while still maintaining descriptive powers. In light of this thinking, a generalized additive model with a smoothing spline based on the X and Y coordinates of the chip was chosen to model the position specific bias.

## 3.3 Creating The Model

When creating the model to correct for background noise, the opportunity to test whether the length of the random probes or the GC content of the random probes had any effect on their signal strength became apparent.  A full model was created that included as factors the length of the probe, the percent of the probe that was either a G or a C, and a spline of the X and Y coordinates.  Only the random probes were ever used as a response variable to reduce the confounding of signals from background noise with signals from the probe actually hybridizing to experimental DNA.

If the data set did not contain random probes, it may have been appropriate to take a similar approach to the MAS 5.0 method and use the probes that represent the bottom two percent of signals in each quadrant.  This will likely include probes that had very little, if any, signal present from the hybridization of actual DNA from the experimental solution.  Therefore, most of the signal from these probes could very well be attributed to background and technical biases.

After the model was created, a likelihood ratio test was performed for each model to test the significance of each factor.  As exemplified by table 3.2, for every model tested, the only significant factor in each case was the spline of the X and Y coordinates.  Each of the other factors, namely length of the probe in base pairs and percent of probes that were either a G or C, were omitted from the final model.  The R environment was used for all analysis in this study.

```
> m = gam(signal ~ perc.GC + lengths + s(X,Y))
> anova(m)

Family: gaussian
Link function: identity

Formula:
signal ~ perc.GC + lengths + s(X, Y)

Parametric Terms:
        df     F p-value
perc.GC  1 0.388   0.534
lengths  1 0.001   0.982

Approximate significance of smooth terms:
         edf Ref.df     F p-value
s(X,Y) 25.65  28.32 8.474  <2e-16
```

**Table 3.2 ANOVA of the full model including GC content, length, and X,Y position:** This particular ANOVA is of the full model constructed for replicate 3. It is evident that the design of the probes removes significance of the GC content and length of the probes as they have insignificant p-values. The X,Y coordinates are clearly significant. Similar results hold in each of the four other replicates.

Given the insignificance of the percentage of GC of the probe as well as the length of the probe, the only factor that was kept in the model was the X and Y coordinates. Therefore, the resulting model looks like the following in R:

```
model = gam(signal ~ s(X,Y))
```

After the model to correct for the position specific background noise has been computed, it is used to remove the biases of the signals. To get an idea of how effective the model was in correcting the background noise, the chi-square values are computed for the corrected data using the same method as was described to test CSR. Most of the new chi-square statistics have increased p-values and many of these new p-values are greater than .05. Therefore, it seems apparent that the model helped move the signals of the random probes to a state of CSR and thus reduce the amount of position specific bias.

# Chapter 4 - Results of Implementing the Model

## 4.1 Evaluative Comparison of Models

In order to judge the effectiveness of the method described in this paper that employs a generalized additive model, it must be compared to the MAS 5.0 as well as the RMA approaches. As was noted earlier in the paper, and is common with most microarray chip, it was apparent that the random probes do not exhibit complete spatial randomness and are thus in need of correction.

To compare the GAM to the MAS 5.0 and RMA models, both methods are employed on the set of five technical replicates of *Triboleum castaneum* data and the results are compared both visually using perspective plots and statistically using the chi-squared statistic to measure complete spatial randomness. Each of the MAS 5.0 calculations were performed using the default settings of 100 for the smoothing factor and the *C* factor set to .5. The lowest two percent of all signals (from both random probes and experimental probes) were used in the MAS 5.0 method as is standard with its algorithm. The RMA calculations were performed by first calculating the mode of the random probe signals, then calculating the standard deviation of the signals in the left tail to estimate $b$, followed by fitting the right tail to an exponential distribution to estimate $a$, and finally by performing the background correction on the random probes.

## 4.2 Chi-square p-value tables

The graphical methods provide a good overall perspective of the effect of each of the models, however, in order to give a numerical measure, tables of the p-values generated by the chi-square statistics were calculated. From these tables it becomes evident that the MAS 5.0 does nothing to establish complete spatial randomness and RMA does very little. On the other hand, in most cases the GAM helps bring the random probe signals to a state of CSR and allows for the further processing of the microarray data.

The GAM is not perfect though and in one particular array, microarray two, the p-values actually worsen. However, this may be due to certain abnormal features of the random probe surface. For future work, the model may be refined to become more robust to these sorts of extreme outliers. What follows is a chip by chip comparison of the background correction methods.

## 4.3 Chip One Visual and Chi-Square Table Discussion

(Figure 4.1) In the first chip the MAS 5.0 seemed to do very little. The visual representation of the random probe signal surface appears to have no change. The RMA model similarly does very little altering, however, it is obviously more than the MAS 5.0 approach. The GAM approach does better than the previous approaches and even helps reduce the effect of the outlying section of very high signaled probes.

(Table 4.1) In terms of results of the chi-square tests, the MAS 5.0 obviously created a floor on the random probes and therefore produced meaningless results for the first quintile and part of the second. The remaining quintiles are identical to the raw data showing no relative change in the signals of the random probes after correction. The RMA results also show very little deviation from the raw data. The GAM approach shows great improvement over the raw data, however, the p-values still do not fall in the range of insignificance in the extreme quintiles.

## 4.4 Chip Two Visual and Chi-Square Table Discussion

(Figure 4.2) In the second chip the MAS 5.0 altered the signals a great deal. The visual representation of the random probe signal surface shows that most of the signals were brought up to a floor while the outlying section of signals did not undergo much alteration. The RMA model shows possibly its most pronounced effect here. The amplification of the main surface is indicative of a change in scale of the graphic indicating that the signals may have been brought much closer to each other. The GAM approach shows a rare case of overcorrection. Most of the surface after the correction looks very smooth, however, in the corner of the outlying signals what once was an outlier in the positive direction is now an outlier in the negative.

(Table 4.2) In terms of results of the chi-square tests, the MAS 5.0 obviously created an even higher floor producing meaningless results for the first three quintiles. The remaining quintiles are identical to the raw data showing no relative change in the signals of the random probes after correction. The RMA method shows the best results here with some improvements, albeit modest, over the original data. These modest improvements are most likely the result of the tendency of RMA to preserve order. In this situation the chi-square results may under represent the actual effect of the correction as evidenced by the visual display. The GAM approach has the uncharacteristic attribute of actually worsening the p-values of the random probes in comparison to the raw data. This may be a result of the portion of overcorrection as well as the chi-square test being insensitive to changes of scale in this scenario.

## 4.5 Chip Three Visual and Chi-Square Table Discussion

(Figure 4.3) In the third chip the MAS 5.0 makes apparent its tendency to produce a flooring effect again while the unfloored region appears to have very little if any change. The RMA model does very little altering and looks almost identical to the raw data. The GAM approach vastly outshines the previous approaches and removes the very obvious shapes in the surface leaving only what looks like random fluctuations.

(Table 4.3) In terms of results of the chi-square tests, the MAS 5.0 is marred by the flooring and has no effect on the upper two quintiles. The RMA results show very little deviation from the raw data which confirms conclusions from the graphic. The GAM approach shows great improvement over the raw data bringing most of the p-values outside of the limits of rejection. The GAM approach seems to really shine with this sort of background.

## 4.6 Chip Four Visual and Chi-Square Table Discussion

(Figure 4.4) In the fourth chip the MAS 5.0 continues to floor the lowest signals and produces a very small effect on the remainder. The RMA model does very little altering and looks almost identical to the raw data with a couple sections exhibiting a small amount of smoothing. The GAM approach again outshines the previous approaches and removes the apparent trends in the surface leaving only smaller patterns near the edges.

(Table 4.4) In terms of results of the chi-square tests, the MAS 5.0 is again dominated by the flooring and has no effect on the upper quintiles. The RMA results again show very little deviation from the raw data which support conclusions from the graphic. The GAM approach shows great improvement over the raw data bringing most of the p-values outside of the limits of rejection only failing to do so in the extreme quintiles and by a relatively small margin. The GAM approach does well again with this sort of background.

## 4.7 Chip Five Visual and Chi-Square Table Discussion

(Figure 4.5) In the fifth chip the MAS 5.0 produces really no apparent effect. The RMA model also yields very little if any effect on the signals. The GAM approach appears to do well for the middle area of signals, however, towards the extremities there may be some overcompensation from the model.

(Table 4.5) In terms of results of the chi-square tests, the MAS 5.0 has no effect on the values outside of the flooring effect in quintile one giving a p-value of zero. The RMA results show very little deviation from the raw data as was seen in the graphic. The GAM approach had almost no positive effect on the p-values of the chi-square test. This was most likely due to the influence of the outliers near the edges have such a dramatic influence on the model.

## 4.8 Concluding remarks

As can be seen by both the graphical displays and the chi-square tables, not only is there evidence for a position specific bias, but the MAS 5.0 method of background correction does nothing to help correct this bias. It does not seem like there is really enough difference in the means of the centroids in order for the correction to have a meaningful effect. What could also be happening is that the smoothing parameter may need adjusting in order for the correction to have more of an impact. In certain situations there is a significant floor in the signals of random probes. This is an indication that the variance of the probe signals is much greater than any differences in the means of the centroids of the sectors. The MAS 5.0 algorithm may perform better if the tuning variables are adjusted completely correctly, however with such a large data set as the microarrays under study this would be a difficult task, especially if it had to be done for each chip individually.

The RMA method on the other hand, may have some effect, however it appears to be very minor and is nearly undetectable using the chi-square test. Therefore, the chi-square test may not be a very appropriate metric by which to judge the algorithm. This is in large part to how the RMA method performs the adjustment. Through analysis of the correction procedure and by statement of the developers of the RMA algorithm, it should be noted that RMA preserves the order of the probe signals that it is adjusting. In particular, as the probe signal being adjusted gets greater, less of an adjustment is being made. The logic in this case is that if a probe exhibits a large signal, that signal is more likely to be a result of accurate DNA hybridization and less likely to be a result of technically based biases.

In contrast to the other methods, a GAM based correction algorithm can have a great deal of effect on the order of the signals of the probes and will therefore show up very well in chi-square tests similar to what was performed here. This is why it is important to consider the perspective plots in judging the effectiveness of the method. One observation that may be worth noting is that the presence of extreme outliers can hinder the performance of the GAM method. The perspective plots reveal a tendency to overcorrect outliers near edges and under-correct outliers that are more centralized on the chip. This is perhaps why some methods mentioned earlier performed clustering analyses in order to identify outliers and make special procedures for them.

Another issue with using the GAM approach is that after the model values have been subtracted from the probes, there may be some probes left with negative signals.  This will cause problems if logarithms are to be used for future manipulation of the signal data.  However, several ways for resolving this issue are available.  For instance, one could simply set the minimum probe signal to a value of one and shift all other probe signals accordingly. Alternatively, those probes which have values below zero likely have no actual hybridization taking place and could therefore likely be omitted without detriment.

Finally, after the technical biases made apparent by the model are removed from the signals of the random probes,  traditional methods for making use of microarray data may be put into practice.  The performance of subsequent analysis techniques will undoubtedly be improved given that much of the noise irrelevant to chemistry of the binding of the probes or the biology of the solution under study has been removed.

For future work, one might focus on how to overcome the challenges presented by the sections of the array that contain extreme signals as these areas were problematic for the GAM. One might also investigate the method of support vector regression mentioned briefly in the literature review as it showed promise of superior results.  Another worthy area of investigation may be performing each of the aforementioned correction methods on experimental data where the concentrations of each of the probe sequences in the solution exposed to the chip was known prior to the experiment.  In this fashion, there would be a very direct way of measuring the effectiveness of each of the algorithms.

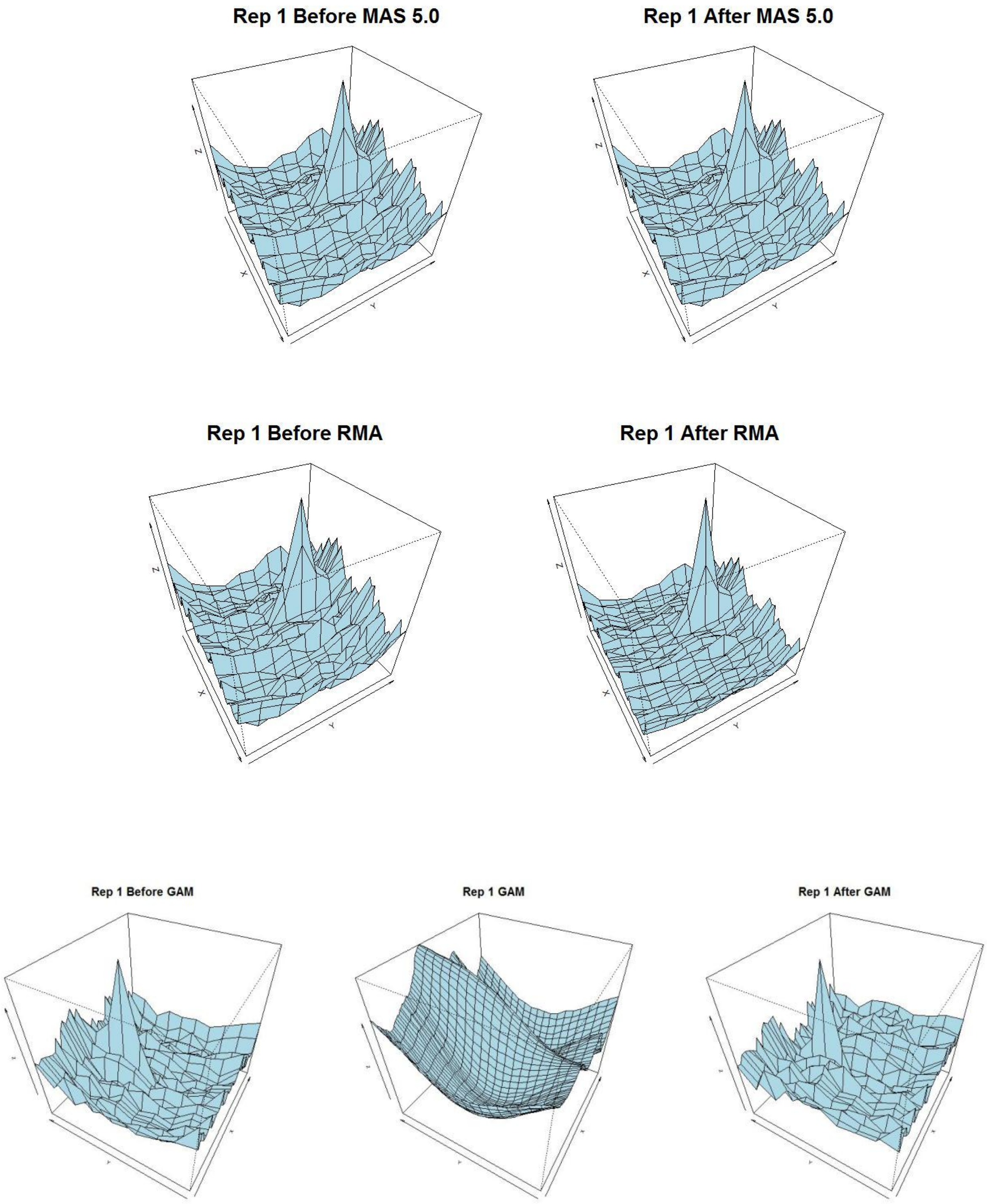


**Figure 4.1 Microarray 1:** The MAS 5.0 model does little to nothing to change the distribution of the signals of random probes. The GAM accounts for increases in signals particularly near the left wall and some near the center.

.

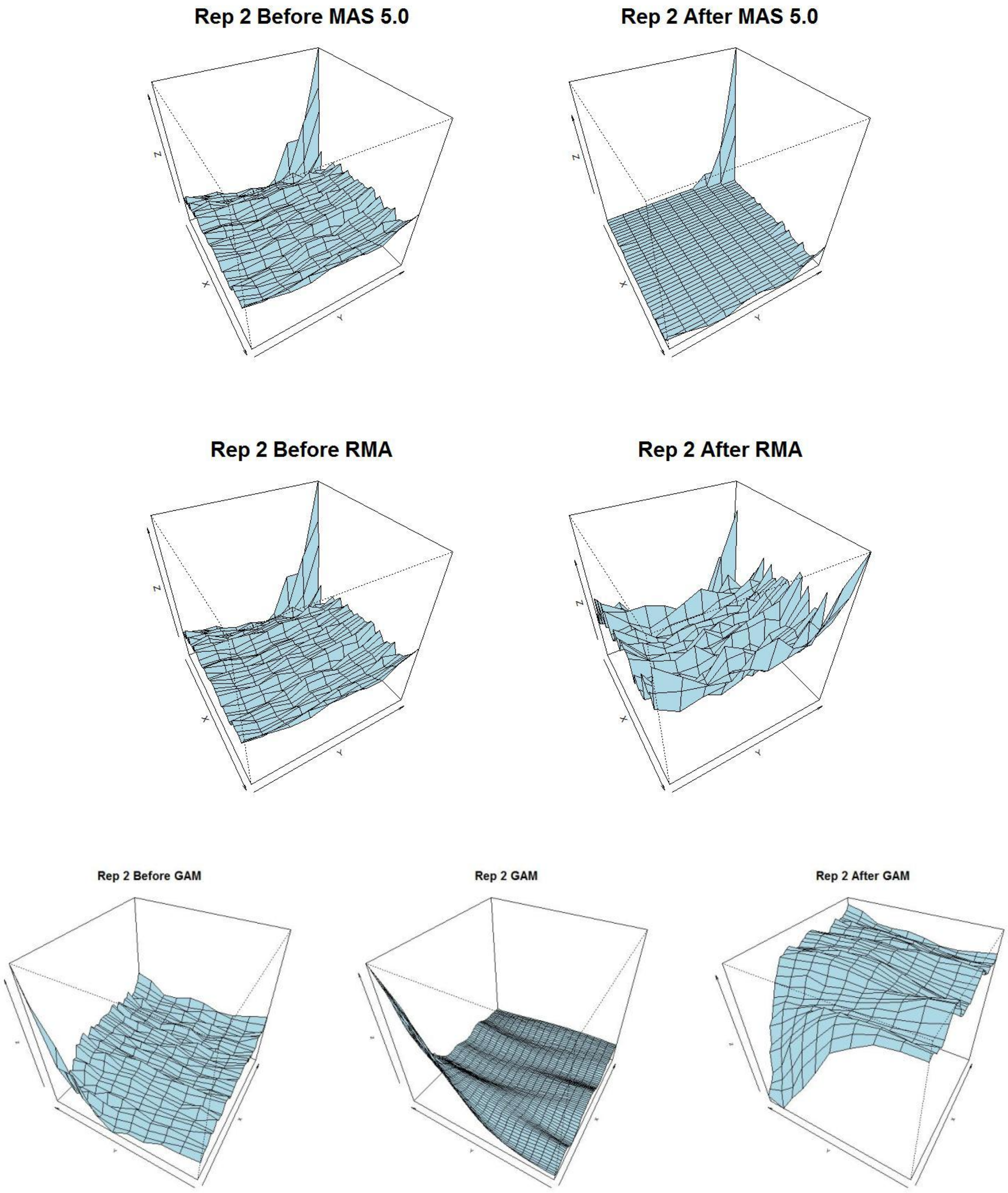


**Figure 4.2 Microarray 2:** The MAS 5.0 model is thrown off by the extreme values found in the corner of this array.  The GAM does a much better job accounting for variation in the middle area of the array, however, it may overcompensate for the extreme corner values.

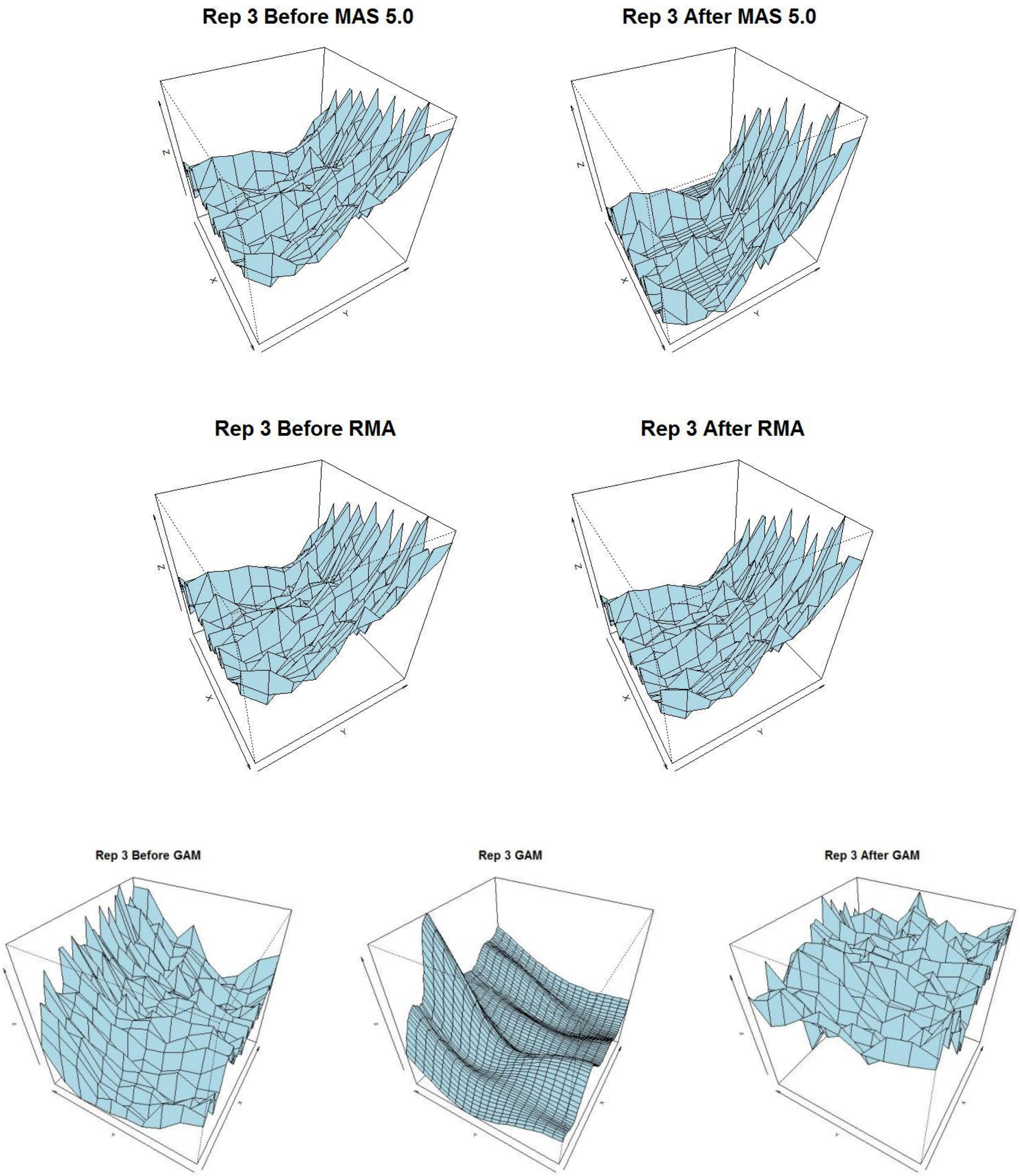


**Figure 4.3 Microarray 3:** The MAS 5.0 method does nothing to account for the obvious bias present towards the edge and simply bring the lowest signal probes up to a minimum value. The GAM does an excellent job in accounting for all the major variation visually present in the random probes.

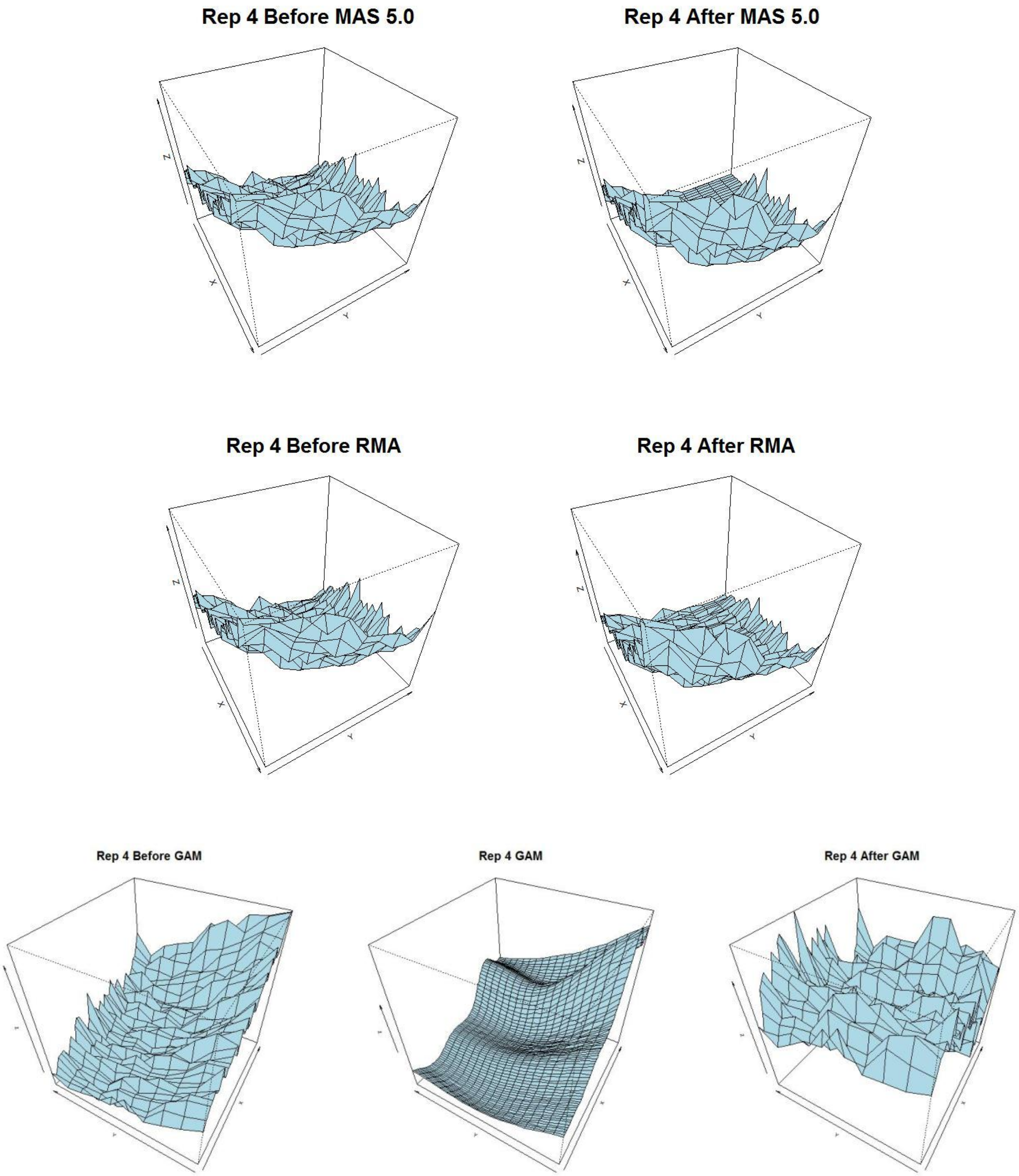


**Figure 4.4 Microarray 4:** The MAS 5.0 method again fails to account for any of the bias obviously present in the random probes. The GAM accounts for the bias very well.

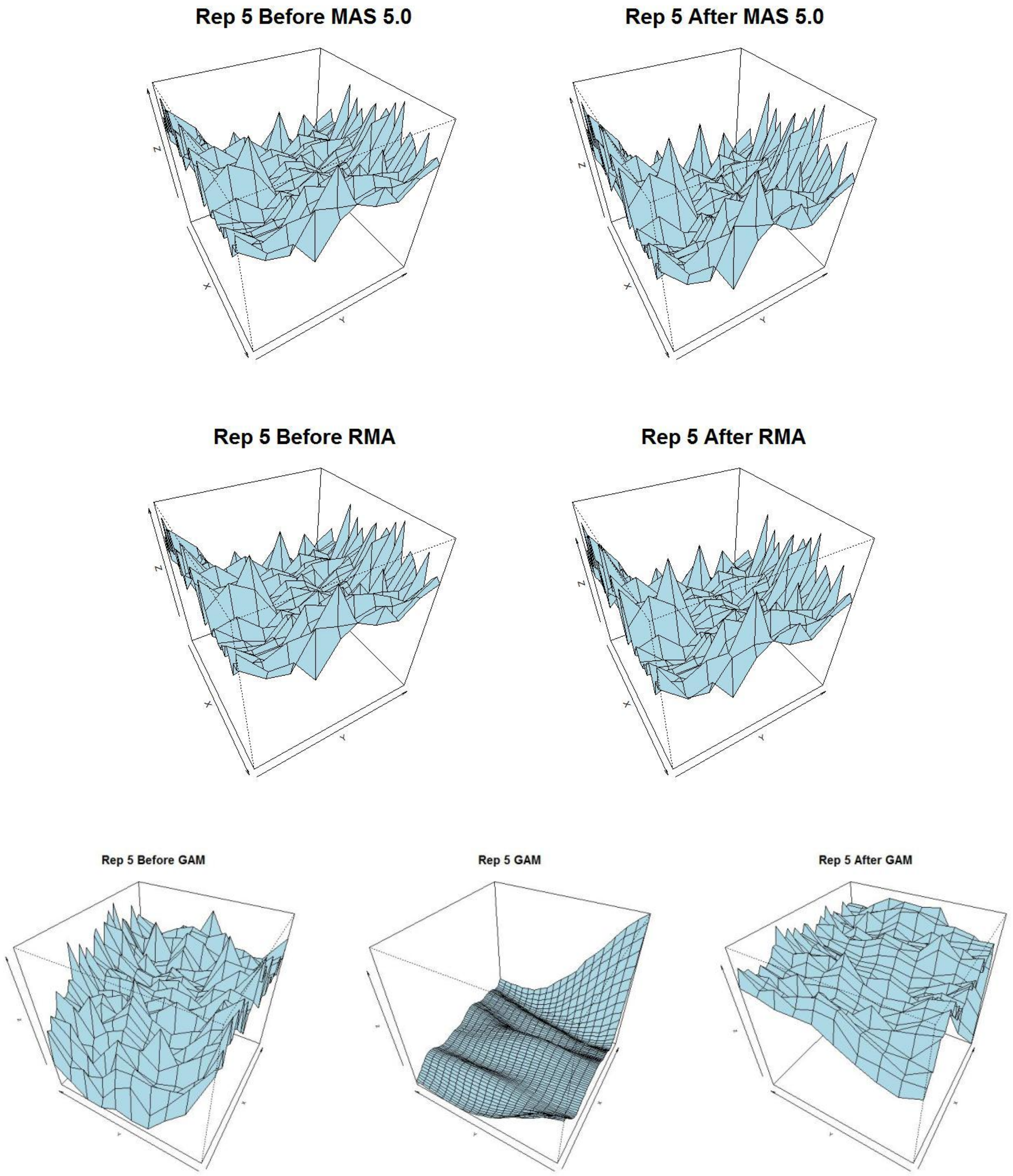


**Figure 4.5 Microarray 5:** The MAS 5.0 method apparently plays no part in modifying the random probe signals. The GAM does well for most of the array, however, it may overestimate the bias in the back right corner.

**Table 4.1 A redisplay of the Chi-Square p-values of the random probes:** This table is redisplayed for the purpose of comparison to the tables after implementation of the MAS 5.0 and GAM methods.

| | Microarray 1 | Microarray 2 | Microarray 3 | Microarray 4 | Microarray 5 |
|---|---|---|---|---|---|
| **1st quintile** | 4.007569e-17 | 1.040798e-39 | 6.987812e-47 | 9.267160e-217 | 7.455458e-36 |
| **2nd quintile** | 1.214135e-02 | 8.596984e-04 | 1.895252e-06 | 1.581560e-43 | 8.628489e-01 |
| **3rd quintile** | 9.860017e-01 | 7.759306e-01 | 9.681052e-01 | 8.810283e-01 | 9.994862e-01 |
| **4th quintile** | 9.643598e-01 | 3.501207e-02 | 8.738846e-02 | 6.244116e-26 | 2.548600e-01 |
| **5th quintile** | 6.093433e-105 | 5.779450e-30 | 2.443781e-114 | 0.000000e+00 | 1.126680e-12 |

**Table 4.2 Chi-Square p-values of the random probes after implementing the MAS 5.0 method:** The MAS 5.0 has no effect on the distribution of the random probes in the upper quintiles. The lower quintiles have a lot of values of zero. This is due to the fact that the MAS 5.0 method simply raises the floor of signal values and sets all low signal probes to the same value. This might be fine if all of the microarrays had a uniform background level, however, as demonstrated earlier, this is not the case.

| | Microarray 1 | Microarray 2 | Microarray 3 | Microarray 4 | Microarray 5 |
|---|---|---|---|---|---|
| **1st quintile** | 0.000000e+00 | 0.000000e+00 | 0.000000e+00 | 0.000000e+00 | 0.000000e+00 |
| **2nd quintile** | 2.118557e-02 | 0.000000e+00 | 0.000000e+00 | 0.000000e+00 | 3.123921e-06 |
| **3rd quintile** | 9.860017e-01 | 0.000000e+00 | 2.500772e-45 | 8.810283e-01 | 9.994862e-01 |
| **4th quintile** | 9.643598e-01 | 1.263174e-37 | 8.738846e-02 | 6.244116e-26 | 2.548600e-01 |
| **5th quintile** | 6.093433e-105 | 5.779450e-30 | 2.443781e-114 | 0.000000e+00 | 1.126680e-12 |

**Table 4.3 Chi-Square p-values of the random probes after implementing the RMA method:** Implementation of the RMA method of background correction generally had very little effect on CSR. Most changes, albeit small, are improvements over the unaltered signals.

| | Microarray 1 | Microarray 2 | Microarray 3 | Microarray 4 | Microarray 5 |
|---|---|---|---|---|---|
| **1st quintile** | 6.133642e-17 | 1.406933e-39 | 3.121427e-46 | 2.807196e-212 | 2.678676e-35 |
| **2nd quintile** | 9.552268e-03 | 5.203667e-04 | 1.365198e-06 | 4.498188e-44 | 8.668379e-01 |
| **3rd quintile** | 9.764845e-01 | 8.563067e-01 | 9.671909e-01 | 9.068782e-01 | 9.987456e-01 |
| **4th quintile** | 9.728165e-01 | 1.077333e-01 | 7.753574e-02 | 2.213666e-24 | 2.898397e-01 |
| **5th quintile** | 1.236481e-107 | 1.867677e-24 | 6.946636e-113 | 0.000000e+00 | 1.028748e-13 |

**Table 4.4 Chi-Square p-values of the random probes after implementing the GAM method:** The p-values displayed in this table are generally greater than the p-values shown in Table 4.1. This is evidence that after implementing the GAM method of background correction, the random probes now display a higher degree of CSR. The only array that does not show improvement in p-values is the second microarray. However, this particular array shows some very extreme outliers in the graphics which may be throwing the model off.

| | Microarray 1 | Microarray 2 | Microarray 3 | Microarray 4 | Microarray 5 |
|---|---|---|---|---|---|
| **1st quintile** | 1.767388e-05 | 5.187979e-72 | 3.231123e-21 | 0.018479701 | 9.776647e-72 |
| **2nd quintile** | 9.975156e-01 | 5.971632e-35 | 9.668313e-01 | 0.996964168 | 8.133016e-01 |
| **3rd quintile** | 9.830104e-01 | 3.869032e-05 | 9.929802e-01 | 0.985850869 | 7.623641e-01 |
| **4th quintile** | 9.900823e-01 | 4.467299e-05 | 6.104833e-01 | 0.999721120 | 4.013092e-01 |
| **5th quintile** | 9.575547e-20 | 1.956509e-09 | 4.948442e-01 | 0.001762096 | 2.031412e-06 |

# References

Baird, D., Johnston, P., Wilson, T. (2004). Normalization of microarray data using a spatial mixed model analysis which includes splines. *Bioinformatics*, 20, 3196–3205

Benito, M., Parker, J., Du, Q., Wu, J., Xiang, D., Perou, C.M., Marron, J.S. (2004). Adjustment of systematic microarray data biases. *Bioinformatics*, 20, 105–114

Bolstad, B., Irizarry, R., Strand, M. and Speed, T. (2002). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*. Vol. 19, Number 2: 185–193

Bolstad, B. (2004). Low Level Analysis of High-density Oligonucleotide Array Data: Background, Normalization and Summarization. *Dissertation.* University of California, Berkeley.

Diggle, P.J. (2003). Statistical Analysis of Spatial Point Patterns (second edition)

Fujita A, Sato JR, Rodrigues Lde O, Ferreira CE, Sogayar MC. (2006). Evaluating different methods of microarray data normalization. *BMC bioinformatics* 2006, 7**:**469.

Hubbell, E. *et al*. (2002) Robust estimators for expression analysis. *Bioinformatics*, 18,1585-1592.

Irizarry, RA, Hobbs, B, Collin, F, Beazer-Barclay, YD, Antonellis, KJ, Scherf, U, Speed, TP (2003) Exploration, Normalization, and Summaries of High Density Oligonucleotide Array Probe Level Data. *Biostatistics* .Vol. 4, Number 2: 249-264

Irizarry, Rafael. A., Bolstad, Benjamin M., Collin, Francois, Cope, Leslie M., Hobbs, Bridget and Speed, Terence P., (2003), Summaries of Affymetrix GeneChip probe level data *Nucleic Acids Research* 31(4):e15

Neuvial P, Hupe P, et. al. (2006) Spatial normalization of array-CGH data. *BMC Bioinformatics* 2006;7:264. doi: 10.1186/1471-2105-7-264.

Wu, Z, Irizarry, RA, Gentleman, R, Martinez Murillo, F, Spencer, F (2004) A Model Based Background Adjustment for Oligonucleotide Expression Arrays. JASA 99(468) 909.

# Appendix A - R code

## A.1 General Custom R Functions Used

```
map.probes = function(data,rep){
        map=matrix(NA,4200,1050)
        for(i in 1:length(data)){
                map[rep[i,"Y"],rep[i,"X"]] = data[i]
        }
        return(map)
}

make.chi.dat = function(dat,blocks){
        d = 1050
        s = ceiling(seq(1,d+1, length.out=blocks+1))
        t = ceiling(seq(1,d*4+1, length.out=blocks*4+1))

        dat = t(matrix(dat, nrow=d))

        vals=numeric()
        for(j in 1:blocks){
                for(i in 1:(blocks*4)){
                        vals = c(vals,sum(dat[t[i]:(t[i+1]-1),s[j]:(s[j+1]-1)], na.rm=TRUE))
                }
        }
        return(vals)
}

make.chi.dat.median = function(dat,blocks, use.mean=FALSE){
        d = 1050
        s = ceiling(seq(1,d+1, length.out=blocks+1))
        t = ceiling(seq(1,d*4+1, length.out=blocks*4+1))

        vals=matrix(NA,blocks*4,blocks)
        for(i in 1:(blocks*4)){
                for(j in 1:blocks){
                        vals[i,j] = ifelse(use.mean==TRUE, mean(dat[t[i]:(t[i+1]-1),s[j]:(s[j+1]-1)],
                                na.rm=TRUE),median(dat[t[i]:(t[i+1]-1),s[j]:(s[j+1]-1)], na.rm=TRUE))
                }
        }
        return(vals)
}

 my.chi.test = function(origdat, dat, percent){
        expect = origdat*percent
        stat = sum((dat-expect)^2/expect)
        return(pchisq(stat, (dim(dat)[1])*(dim(dat)[2]), lower.tail = FALSE))
}
```

## A.2 Loading in the Data and Setting General Parameters

```
nrandom = 30938

data.dir <- "C:\\Users\\Dustin\\Desktop\\Tiling Arrays\\Pairs\\Five_Tech_Reps_one-color"
setwd(data.dir)

###	Read in one example pair file for reference of probe locations
reps.names = list.files(pattern = "48-60hr_embryonic_rep1_532.pair")
rep=read.table(reps.names, sep="\t", header=T)



###	Read in the previously written table containing just the signal strengths from the gff3s
original.dat = read.table("original dat.txt", header=FALSE, sep = "\t")
random.dat = original.dat[1:nrandom,]

new.data=original.dat
dim(new.data)
dim(rep)
numofbins = 400
xsplit = round(sqrt(numofbins/4))
```

## A.3 Performing Analysis on the Raw Data

```
####	mapping, then grouping and testing the probes
####	checking to see how well the data is distributed
rep.map = map.probes(new.data[1:nrandom,1], rep)
presence = rep.map > 0
p.visual = matrix(make.chi.dat(presence,xsplit), ncol=xsplit)
my.chi.test(matrix(mean(p.visual),nrow=dim(p.visual)[1],ncol=dim(p.visual)[2]),p.visual,1)
mean(p.visual)
p.visual


####	mapping, then grouping and testing the random probes for CSR
####	choose i to be the chip number (1-5 for tech reps); k is the number of quantiles
k=5; ri.map=list(); ri.visual=list(); chi=matrix(1,nrow=k,ncol=5)
for(i in 1:5){
	ri.map[i] = list(map.probes(new.data[1:nrandom,i], rep))
	for(j in 0:(k-1)){
		ri = (unlist(ri.map[i]) > quantile(new.data[1:nrandom,i],j/k))* (unlist(ri.map[i])
			< quantile(new.data[1:nrandom,i],(j+1)/k))
		ri.visual[((i-1)*k)+(j+1)] = list(matrix(make.chi.dat(ri,xsplit), ncol=xsplit))
		chi[(j+1),i] = my.chi.test(p.visual, matrix(unlist(ri.visual[((i-1)*k)+(j+1)]),ncol=xsplit), 1/k)
	}
}
chi
```

## A.4 Performing the MAS 5.0 Analysis

```
mas.5 = function(back.map, blocks){
        d = 1050
        s = ceiling(seq(1,d+1, length.out=blocks+1))
        t = ceiling(seq(1,d*4+1, length.out=blocks*4+1))

        # In the means.coords array the first value is the mean of the quadrant, the second is the
        #variance, the third is the X address of the centroid, and the fourth is the Y address of the
        #centroid.
        means.coords = array(NA, c(blocks*4,blocks,4))

        for(i in 1:(blocks*4)){
                for(j in 1:blocks){
                        section = as.vector(back.map[t[i]:(t[i+1]-1),s[j]:(s[j+1]-1)])
                        section = na.omit(section)
                        section = section[(section< quantile(section,.02, na.rm=T))]
                        means.coords[i,j,1] = mean(section)
                        means.coords[i,j,2] = var(section)
                        means.coords[i,j,3] = (s[j]+s[j+1]-1)/2
                        means.coords[i,j,4] = (t[i]+t[i+1]-1)/2
                }
        }
        return(means.coords)
}

recalc.probes = function(map ,rep, mas.info){
        # map is a matrix where nrow=4200 and ncol=1050 representing the chip cells
        weight = matrix(NA, nrow = dim(mas.info)[1], ncol = dim(mas.info)[2])

        for(k in 1:nrandom){
                for(i in 1:dim(mas.info)[1]){
                        for(j in 1:dim(mas.info)[2]){
                                weight[i,j] = 1/(((rep[k,"X"]-mas.info[i,j,4])^2+(rep[k,"Y"]-
                                        mas.info[i,j,4])^2)^.5+100)
                        }
                }
                weight[i,j] = weight[i,j]/sum(weight[i,j])

                map[rep[k,"Y"],rep[k,"X"]] = max(map[rep[k,"Y"],rep[k,"X"]]-mas.info[i,j,1]*weight[i,j],
                        .5*sqrt(mas.info[i,j,2])*weight[i,j])
        }
        return(map)
}

map.array = array(NA, c(4200,1050,5))
map.recalc.array = array(NA, c(4200,1050,5))
k=5; mas.chi = matrix(0,k,5)
mas.ri.visual = list()

for(i in 1:5){
        background.map = map.probes(new.data[,i], rep)
        map = map.probes(new.data[1:nrandom,i], rep)
```

```
          mas.info = mas.5(background.map,xsplit)
          map.recalc = recalc.probes(map, rep, mas.info)

          map.array[,,i] =  map
          map.recalc.array[,,i] = map.recalc

          for(j in 0:(k-1)){
                    ri = (map.recalc > quantile(map.recalc,j/k, na.rm=T))* (map.recalc <
                              quantile(map.recalc,(j+1)/k, na.rm=T))
                    mas.ri.visual[((i-1)*k)+(j+1)] = list(matrix(make.chi.dat(ri,xsplit), ncol=xsplit))
                    mas.chi[(j+1),i] = my.chi.test(p.visual, matrix(unlist(mas.ri.visual[((i-
                              1)*k)+(j+1)]),ncol=xsplit), 1/k)
          }
}
setwd("C:\\Users\\Dustin\\Desktop")
save.image()

mas.chi
```

## A.5 Visualization of the MAS 5.0 Background Correction

```
for(i in 1:5){
          original = make.chi.dat.median(map.array[,,i],xsplit)
          corrected = make.chi.dat.median(map.recalc.array[,,i],xsplit)

          #jpeg(paste("Rep",i,"Before and After MAS 5.0.jpeg"), w=1200, h=600)
          par(mfrow=c(1,2))
          persp(original, phi=40,theta=60, col="light blue",
                    main = list(paste("Rep", i, "Before MAS 5.0"), cex=2.5), xlab = "X")
          persp(corrected, phi=40,theta=60, col="light blue",
                    main = list(paste("Rep", i, "After MAS 5.0"), cex=2.5), xlab = "X")
          #dev.off()
}
```

## A.6 Performing the RMA Analysis

```
RMA.vals = matrix(0,nrandom,5)
log.random.dat=log(random.dat,2)
library(MASS)

for(i in 1:5){
          fit = density(log.random.dat[,i])
          xmode = fit$x[fit$y ==  max(fit$y)]
          plot(fit, xlim=c(7,11))
          abline(v=xmode, col="red")
          mu = xmode
          left.values = log.random.dat[log.random.dat[,i]<xmode,i]
          right.values = log.random.dat[log.random.dat[,i]>xmode,i]
          length(left.values); length(right.values)

          sstd = sqrt(sum((left.values-mu)^2)/(length(left.values)-1))
          theta = sstd
          b = theta

          alpha = as.numeric(fitdistr(right.values-mu, "exponential")$estimate)

          for(j in 1:nrandom){
                    a = log.random.dat[j,i] - mu - theta^2 * alpha
                    y = function(x) (x/b)*dnorm((x-a/b))

                    RMA.vals[j,i] = (1/(pnorm(a/b)+pnorm(log.random.dat[j,i])-1))*integrate(y , 0,
                              log.random.dat[j,i])$value

          }
}

map.array = array(NA, c(4200,1050,5))
RMA.map.array = array(NA, c(4200,1050,5))
k=5; rma.chi = matrix(0,k,5)
rma.ri.visual = list()
for(i in 1:5){
          map.array[,,i] = map.probes(random.dat[,i], rep)
          RMA.map.array[,,i] = map.probes(RMA.vals[,i], rep)

          for(j in 0:(k-1)){
                    ri = (RMA.map > quantile(RMA.map,j/k, na.rm=T))* (RMA.map <
                              quantile(RMA.map,(j+1)/k, na.rm=T))
                    rma.ri.visual[((i-1)*k)+(j+1)] = list(matrix(make.chi.dat(ri,xsplit), ncol=xsplit))
                    rma.chi[(j+1),i] = my.chi.test(p.visual, matrix(unlist(rma.ri.visual[((i-
                    1)*k)+(j+1)]),ncol=xsplit), 1/k)
          }
}
rma.chi
```

## A.7 Visualization of the RMA Background Correction

```
for(i in 1:5){
          original = make.chi.dat.median(map.array[,,i],xsplit)
          corrected = make.chi.dat.median(RMA.map[,,i],xsplit)

          #jpeg(paste("Rep",i,"Before and After MAS 5.0.jpeg"), w=1200, h=600)
          par(mfrow=c(1,2))
          persp(original, phi=40,theta=60, col="light blue",
                    main = list(paste("Rep", i, "Before MAS 5.0"), cex=2.5), xlab = "X")
          persp(corrected, phi=40,theta=60, col="light blue",
                    main = list(paste("Rep", i, "After MAS 5.0"), cex=2.5), xlab = "X")
          #dev.off()
}
```

## A.8 Performing the GAM Analysis

```
library(mgcv)

randomdata =data.frame(X=rep$X[1:nrandom],Y=rep$Y[1:nrandom], Rep1=new.data[1:nrandom,1],
                         Rep2=new.data[1:nrandom,2], Rep3=new.data[1:nrandom,3],
                         Rep4=new.data[1:nrandom,4], Rep5=new.data[1:nrandom,5])

preds=matrix(0,nrow=nrandom,ncol=5)
for(i in 1:5){
                         m =gam(randomdata[,i+2]~s(X,Y),data=randomdata )
                         preds[,i]=predict(m)
                         #print(anova(m))
}

####                     mapping, then grouping and testing the corrected probes for CSR
####                     choose i to be the chip number (1-5 for tech reps); k is the number of quantiles
k=5; ri.visual = array(NA, c(40,10,25)); chi.aftermodel=matrix(1,nrow=k,ncol=5)
for(i in 1:5){
                         for(j in 0:(k-1)){
                              ri = (correct.map[,,i] > quantile(correct.map[,,i],j/k, na.rm=TRUE))*
(correct.map[,,i] < quantile(correct.map[,,i],(j+1)/k, na.rm=TRUE))
                              ri.visual[,,((i-1)*k)+(j+1)] = make.chi.dat(ri,xsplit)
                              chi.aftermodel[(j+1),i] = my.chi.test(p.visual, ri.visual[,,((i-1)*k)+(j+1)], 1/k)
                         }
}
chi.aftermodel
```

## A.9 Visualization of the GAM and the Background Correction

```
rand.map=array(NA, c(4200,1050,5)); pred.map=array(NA, c(4200,1050,5)); correct.map=array(NA,
c(4200,1050,5))
for(i in 1:5){
          rand.map[,,i] = map.probes(randomdata[,i+2], rep)
          pred.map[,,i] = map.probes(preds[,i], rep)
          correct.map[,,i] = map.probes(randomdata[,i+2]-preds[,i], rep)
}
setwd("C:\\Users\\Dustin\\Desktop\\School\\Master's Report")
for(i in 1:5){
          original = make.chi.dat.median(rand.map[,,i],xsplit)
          model = make.chi.dat.median(pred.map[,,i],xsplit*2)
          corrected = make.chi.dat.median(correct.map[,,i],xsplit)

          #jpeg(paste("Rep",i,"Before, Model, and After GAM.jpeg"), w=1800, h=600)
          par(mfrow=c(1,3))
          persp(original, phi=40,theta=-60, col="light blue",
                    main = list(paste("Rep", i, "Before GAM"), cex=2.5), xlab = "X")
          persp(model, phi=40,theta=-60, col="light blue",
                    main = list(paste("Rep", i, "GAM"), cex=2.5), xlab = "X")
```

```
		persp(corrected, phi=40,theta=-60, col="light blue",
			main = list(paste("Rep", i, "After GAM"), cex=2.5), xlab = "X")
		#dev.off()
	}
```