The Advantages of a Biweight Metric in Clustering Microarray Data

Robert Kurtzman

Pomona College

Advisor: Johanna Hardin

Department of Mathematics

Spring 2008

Contents

List of Tables

List of Figures

Abstract

Distance metrics are often the backbone of clustering algorithms. Yet certain distance metrics, such as one based on Pearson's correlation, are sensitive to outliers. Microarray data tend to have outlying data points. Hence, we may intuitively believe metrics like one based on Pearson's correlation may not be appropriate for clustering microarray data. Hardin, et al. (2007) show a metric based on Tukey's biweight estimate of multivariate scale and location to be more robust than a metric based on Pearson's correlation. The goal of this paper is to find a way to evaluate whether a metric based on Tukey's biweight will create more valid clusters (based on known partitions in microarray data) than a metric based on Pearson's correlation.

We define an extreme outlier to be a data point that is sufficiently outlying with respect to the relationship between two genes. When an extreme outlier is removed, a metric based on Pearson's correlation will often dramatically change its measurement of correlation, while a metric based on Tukey's biweight will often display very little change in its estimate of correlation [11]. We consider a "robust" cluster to be a cluster that would be clustered the same with the removal of extreme outliers from gene pairings as without the removal. Our results suggest that a metric based on Tukey's biweight will create more "robust" clusters than a metric based on Pearson's correlation. As our clustering algorithm, we use Partitioning Around Medoids (PAM) [14].

Chapter 1

Background: Introduction to Clustering Microarrays

Why is one person afflicted with lung cancer while another person is not? We often read that there are paramount differences between how the two people lived their lives. We have also come to see that genetic influences can be integral to determining a person's chances of getting cancer. As scientists, we want to get to the heart of the subject, and find how the two subjects' body chemistries differ.

The most useful building blocks for scientists are genes. Genes are segments of our DNA; hence, each gene's information is a part of our genetic blueprint. Genes live in two states, active (or "expressed") and inactive. A gene is active when the information on the gene is being "read." During a well-known multiple-stage biological process known as transcription and translation, the "reading" of the information in the gene occurs. The result of transcription and translation is often the creation of a protein. Proteins perform specialized functions within cells.

In the process of transcription, a molecule known as messenger RNA (mRNA) is created to make a copy of certain information on the gene. The mRNA resulting from this process is called an mRNA transcript. This mRNA carries this information to the sites where protein synthesis occurs. At these sites, the information from the mRNA is often used to create a protein by the process of translation.

The level of gene activity (or its expression level) is usually defined as the number of mRNA transcripts synthesized by a gene to conduct the synthesis of different proteins. Thus, to analyze gene expression, scientists compare the relative quantities of various mRNA molecules in the cells of two different subjects.

In 1989, Affymetrix, invented a technology that can track the activity of thousands genes. The technology Affymetrix invented, known as a microarray, is a small robotic chip. The microarray is a matrix of spots, i.e., locations that will be used to measure a certain gene's expression level. There are a number of different microarray chips on the market today. Note, the method we describe below for using a microarray to analyze gene expressions is not the method used by Affymetrix.

To study an organism, scientists lyse the cells they are interested in, and then they isolate the mRNA. They then convert the mRNA into a complementary strand known as complimentary DNA (cDNA). Each spot on the microarray contains a complement of the cDNA, to which the cDNA can bind. The scientists spread the cDNA onto the chip to measure whether genes are expressed.

Our specific problem is that of finding groups of genes that are co-expressed. To study differences in co-expression between two human subjects, the chip will have all the genes that can feasibly be thought to be active in our subjects. Then, we will isolate the mRNA from the tissue we are interested in from the healthy and unhealthy subjects. The cDNA is created and is then labeled red if it comes from the unhealthy subject and green if it comes from the healthy subject. The cDNA is then spread across the microarray, and the genes that would be "expressed" by the mRNA on the chip light up the color of the cDNA.

When genes are active in both healthy and unhealthy subjects, the displayed color is yellow. When the gene is inactive in both subjects, the displayed color is black. The colors are then quantified; because sometimes the cDNA only binds partially with the spot, we get a full spectrum of values. In fact, there is almost never a time when a spot on the chip lights up to be completely red or green. There are more cases where the spot is completely black. The quantified value is the log ratios of the color intensities of each spot. We use standardized values. Each microarray gives us one array worth of data for thousands of genes. We repeat this process for as many arrays as possible or necessary.

1.1 The Goal of Working with A Microarray:

Our data is a matrix of rows of thousands of genes, and columns of tens of arrays. The data is used to see which genes are correlated with other genes. If we can find large groups of genes that are all correlated with one another then we have clusters. Hence, an analysis of the clusters can effectively find groups of genes that are correlated with one another. Therefore, if we have two genes in the same cluster, we know that these two genes are expressed across the different arrays in such a way that they are co-expressed by some measure. Likewise, if we have fifty genes that are all correlated, these genes are supposedly all active to a similar degree across different arrays. In our analysis, we consider to what degree genes in the same cluster are active. By identifying patterns of gene expression and grouping genes into expression classes, we might be able to provide much greater insight into their biological function and relevance.

Chapter 2

Clustering Techniques: PAM

Clustering algorithms perform the intuitively simple task of finding groups in data. Distance metrics are the backbone of these algorithms because a group is going to be defined by the proximity of data points to a certain point. We look at distance metrics in the next chapter. We examine unsupervised learning techniques, since we have no *a priori* knowledge about the cluster structure.

Clustering algorithms tend to be categorized into two types: partitioning and hierarchical. There are also hybrids of these two types of algorithms as well as other types of algorithms. Partitioning Around Medoids (PAM) is a partitioning algorithm. We use this algorithm in our research. The algorithm performs the following steps:

•Choose the number of clusters you want, say k.

•Choosing Medoids (BUILD).
–Select the first medoid by choosing the data point for which the sum of all the dissimilarities to all other elements is as small as possible.
–To Select the remaining k −1 medoids:

1. Consider an element, i, which has not yet been selected.
2. Consider another non-selected element, j.
3. Calculate the difference between the dissimilarity, D_j, of element j with the most similar previously selected medoid and the element's dissimilarity, $d(j, i)$, with element i.
4. If the difference from Step 3 is positive, element j will benefit if element i is selected as the medoid. We calculate:

$$C_{j,i} = \max(D_j - d(j, i), 0) \tag{2.1}$$

5. We calculate the total gain by selecting element i:

$$\sum_j C_{j,i}$$

6. Choose as the next medoid the not yet selected element i which maximizes the total gain:

$$\max{}_i \sum_j C_{j,i}$$

–Continue steps 1-6 until k medoids have been found.

•See if an unselected element may be a better choice as a medoid than the current selections (SWAP).

1. Consider a non-selected object, h, along with our current selected object, i. Also, consider another representative object, j, and calculate its contribution to the swap, C_{jih}. Let $C_{jih} =$

   (a) Zero if j is further from both i and h than from another representative object.

   (b) One of two values defined below if j is not further from i than from any other medoid. Note:

   $$D_j = d(j, i) \tag{2.2}$$

   i.

   $$C_{jih} = d(j, h) - D_j \tag{2.3}$$

   if j is closer to element h than to the second closest medoid, i.e., $d(j, h) < E_j$ where E_j is the dissimilarity between j and the second most similar representative object. In this case, the contribution of element j to the swap between objects i and h.

   ii.

   $$C_{jih} = E_j - D_j \tag{2.4}$$

   if j is at least as distant from h than from the second closest medoid, i.e., $d(j, h) > E_j$.

   (c)

   $$C_{jih} = d(j, h) - D_j \tag{2.5}$$

   if j is further from medoid i than from at least one of the other representative objects but closer to element h than to any medoid.

2. Calculate the total results by adding the contributions:

$$T_{i,h} = \sum\nolimits_j C_{jih} \tag{2.6}$$

–To decide whether to carry out a swap:

1. Select the pair *(i,h)* which minimizes $T_{i,h}$.

2. If the minimum of $T_{i,h}$ is negative, the swap is carried out and the algorithm returns to step 1. If the minimum of $T_{i,h}$ is positive or 0, the value of the objective cannot be decreased by carrying out a swap and the algorithm stops.

2.1 Understanding PAM and PAM's Advantages

In PAM, we must input the number of clusters we believe there are in our data. We will show how we can avoid this in the chapter, "Methods and Results." We then try to find medoids. There will be as many medoids as clusters. PAM finds the most representative medoids in our data by calculating the within-group distances of all the possible combinations of medoids. We want groups that have the smallest within-group distances from each other, and the furthest distances from the other groups. Intuitively, it makes sense that we want our data to be as close as possible to one another if they are in a group, and as far away as possible from the points in the other groups.

We do not measure the validity of the clusters PAM creates relative to clusters created by other clustering algorithms. In the literature, clustering algorithms have been studied, and although there is no consensus on a "best" algorithm, partitioning and hybrid methods have been shown to create more valid clusters (based on known partitions within the data) than hierarchical methods [2], [23]. We clustered our data with other algorithms like HOPACH and a linked-hierarchical method, but we found that the output from PAM was the best means to studying our different metrics.

Chapter 3

Distance Metrics

To be a distance metric, the metric, d, must satisfy the following four properties for some points x, y, and z.

$$\begin{aligned} d(x,y) &\geq 0 && (3.1) \\ d(x,y) &= d(y,x) && (3.2) \\ d(x,y)=0 &\iff x=y && (3.3) \\ d(x,y) &\leq d(x,z)+d(z,y) && (3.4) \end{aligned}$$

In $\Re^2$, distance metrics define a distance between some $x \in \Re^2$ and some $y \in \Re^2$. Likewise, in $\Re^n$, distance metrics define a distance between some $x \in \Re^n$ and some $y \in \Re^n$. A correlation measure suggests the strength and direction of a linear relationship between two random variables in some space. There are a number of distance metrics that are functions of correlation measures. A distance metric in $\Re^n$ that is a function of a correlation measure estimates the distance between some $x \in \Re^n$ and some $y \in \Re^n$. A high correlation between these two points implies a small distance. Likewise, a low correlation between x and y implies a large distance.

In our project, we examine thousands of correlations between different genes using distance metrics based on correlation measures, and we group together highly correlated genes. We considered a few different metrics based on correlation coefficients, specifically based on Pearson's correlation, Tukey's biweight correlation, and Spearman's correlation. We use two other distance metrics to simulate clusters: the Euclidean metric and the absolute Euclidean metric. Note that there are many metrics that sustain the above properties, 3.1-3.4; we merely consider the metrics most used in clustering microarray data. Also note that equations 3.3 and 3.4 are not always satisfied by the metrics based on correlation. This is acceptable to our study. The metrics based on correlation may not be "distance metrics," because they do not satisfy the properties 3.1-3.4. However, metrics based on correlations are useful to clustering algorithms - and, in turn, to our study - because they estimate closeness based on correlation.

Below, we examine the different distance metrics in depth.

3.1 The Euclidean and Absolute Euclidean Metric

We begin with the Euclidean metric:

$$d_E(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{3.5}$$

The Euclidean metric satisfies equations 3.1-3.4. The metric is not considered to be robust.

The absolute Euclidean metric is:

$$d_{abs(E)}(x,y) = \min(d_E(x,y), d_E(-x,y)) \tag{3.6}$$

The absolute Euclidean metric considers two distances and takes the minimum. The first distance it considers is the Euclidean distance above. The other distance it considers is the Euclidean distance metric after a minus transformation on one of the components, x or y.

The absolute Euclidean metric is not a well-known metric, but it is pertinent for the analysis of microarray data. Since each data point is the log ratio of two values, a minus transformation on a data point will be the log ratio of the inverse of the two values. In our case, if genes x and y are close, we want to cluster them; likewise, if genes x and $-y$ are close, we want to cluster the two genes. Hence, the absolute Euclidean metric is pertinent to our specific analysis of microarray data.

3.2 A Metric Based on Pearson's Correlation Coefficient

A metric based on Pearson's correlation is a function of Pearson's correlation. The equation for the metric based on Pearson's correlation is:

$$d_r(x,y) = 1 - |r_{xy}| \tag{3.7}$$

where r_{xy} is the Pearson's correlation coefficient of the vectors x and y.

$$r_{xy} = \frac{S_{xy}}{\sqrt{S_x}\sqrt{S_y}} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{3.8}$$

Pearson's correlation coefficient, r_{xy}, is the product of the differences in the mean deviations divided by the product of the standard deviations. Correlation is defined as the ratio of the covariance to the product of the individual standard deviations. Hence, all further correlation measures are variations on the idea of dividing the covariance by the product of the standard deviations.

3.3 A Metric Based on Spearman's Rank Correlation Coefficient

The Spearman correlation coefficient is the Pearson's correlation coefficient on the ranked data. The formula below is the commonly used formula for the Spearman correlation:

$$d_S(x, y) = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \tag{3.9}$$

where d_i = the difference between the rank of each corresponding value of x and the rank of each corresponding value of y.
n = the number of pairs of values

Two cases define Spearman's rank correlation coefficient: there are tied ranks and there are no tied ranks. When there are no tied ranks, we perform our calculation as in equation 3.9.

If tied ranks exist, we first rank the data using the average of the two ranks any two tied points would have been. For example, say we have two equal data points. Within our vector, x, the equal data points are the fourth and fifth biggest points. Hence, we assign them each rank 4.5. We then perform our calculation as in equation 3.9.

The metric used in our paper is:

$$d_S(x, y) = 1 - |r_S(x, y)| \tag{3.10}$$

There are some clear differences between the formulas for the Spearman's and Pearson's coefficients. Spearman's coefficient does not require that we think of the relationship between x and y as linear. Likewise, we measure from an ordinal rather than a cardinal perspective. In data with many outliers, these differences are often seen as advantages. However, we do see inherent problems with Spearman's coefficient. It ignores the information about relative distance between data points. Hence, we consider another robust distance metric, a metric based on Tukey's biweight correlation.

3.4 A Metric Based on Tukey's Biweight Correlation

A metric based on Tukey's biweight is considered "resistant"to outliers. The biweight correlation comes from the class of M-Estimators, which have been shown to down-weight points that are far from the estimated center as defined by the scatter of the data [13]. To make good choices about which points to down-weight, we need a good estimate of the center of the data and how the data is scattered. It is important to note that sometimes outliers are in fact important and meaningful points. A valuable part of the metric based on the biweight correlation is that it gives the user the ability to determine whether points should be flagged (and perhaps later removed) as outliers.

Consider two genes. We can output a 2x2 biweight covariance matrix from the biweight measurement of multivariate scale:

$$\begin{pmatrix} s_{BWx,x} & s_{BWx,y} \\ s_{BWx,y} & s_{BWy,y} \end{pmatrix}$$

Using our definition of correlation, we use the output from our 2x2 biweight covariance matrix to define the biweight correlation:

$$r_{BW} = \frac{s_{BWx,y}}{\sqrt{s_{BWx,x} * s_{BWy,y}}} \tag{3.11}$$

In turn, we define the metric based on biweight to be:

$$d_{BW} = 1 - |r_{BW}| \tag{3.12}$$

3.5 A Note on the Specification of Metrics Throughout this Paper

We have called a metric based on a correlation "a metric based on Pearson's correlation," for example. Now that we have a good understanding of the metrics based on correlation, we shall call, throughout the rest of the paper, the metrics based on correlation which we use: Pearson's metric, Spearman's metric, and the biweight metric.

Chapter 4

Methods: Simulating and Clustering Data

4.1 Data

Our data is a matrix of yeast microarray gene expression data: 3360 genes (rows), 17 arrays (columns). The arrays represent samples taken over multiple different generations.

Clearly, we have a large number of genes, but we do not have a large number of data points, per se, because in our correlations between two genes we have at most 17 data points.

4.2 Simulating Data

A clustering approach is considered more valid if the algorithm partitions the data such that the algorithm is closer (by some measure) to creating the known partitions. Our goal is to compare the biweight and Pearson's metrics by distinguishing between the validity of the biweight and Pearson's metric's re-clusterings of the simulated clusters.

To accomplish our goal, we create a simulated sample from our data. We created "large" clusterings of 5 groups of 50 genes and "small" clusterings of 3 groups of 20 genes. A subset of genes with small within cluster distances were selected in creating 3 or 5 clusters. The metrics used for simulation were: absolute Euclidean, Euclidean, Spearman's, Pearson's, and biweight.

To simulate data:

1. Select a single gene, x, at random. This gene is called a *node*.
2. Create a one-column matrix, M, with same number of rows as one less than the total number of genes (in our case, total genes = 3360, so number of rows = 3359).

3. In each row, insert the distance (as determined by the specific distance metric) from x to a specific gene in the sample, with no repetition of the gene selected.

4. Create a new one-column matrix, N, in which the rows are an ordered list of the distances in M from lowest to highest (closest to furthest from x).

5. Select the 19 or 49 closest genes to x (depending on the size of the cluster wanted), and put these genes in the cluster.

6. Remove the rows of the distances of the genes that were selected to be in the cluster from M.

7. Select the minimum of the list (the furthest gene) from our initially selected gene, x'. This gene is another *node*.

8. Remove the distance from x to x' from M.

9. Create a one-column matrix, M', in which each row is the distance from x' to some other gene in the data besides the genes already selected to be in the created cluster(s) (again, with no repetition of the gene selected). The distance from x' to some gene, say a, should be in the same row as the distance from x to a.

10. Create a two-column matrix by attaching M' to M. Call this matrix M, replacing the previously created M.

11. Create a one-column matrix of the minimum value of each row in M (i.e., find the smallest of the distances to each selected node from each row). Call this matrix N, replacing the previously created matrix N.

12. Select the 19 or 49 genes with the lowest values in N. Put these genes in the new cluster around the node x'.

13. Remove the 19 or 49 selected genes' distances from M.

14. Select the gene with the highest value in N. Take this gene as a new node, around which to form, a new cluster. Call this gene x', replacing the previously created x'.

15. Remove the distances from x' to the previously selected nodes from M.

16. Repeat steps 9-15 until one less than the number of clustered wanted are created. Then repeat steps 9-12.

It is important to note that we did not allow genes to be put in more than one group, i.e., all selection was done without replacement.

Our chosen method for simulating the data is problematic. Just because our initial node, chosen in step 1, and and our next chosen node, chosen in step 7, are far from one another, this does not mean that the points within the two node's clusters, chosen in step 5, are all necessarily far from one another. Hence, for any metric used in the simulation process, we do not believe that the simulated clustered are objectively true clusters. Thus, it is not possible to attain any notion of the validity of the clustering technique by measuring the ability of the clustering technique to re-cluster simulated clusters into the original partitions.

At first, it was unclear if the problems stated above would make our results moot. However, we were still able to approach our goal of examining differences in how biweight and Pearson's cluster simulated data, as will be explained below.

4.3 Clustering

When re-clustering our simulated clusters, we do not want to make any assumptions about the specific number of clusters our clustering techniques will think is best. Rather, we use a reflection of the appropriateness of the number of clusters assigned, i.e., average silhouette width, to choose the appropriate number of clusters. We explain silhouette width below. The average silhouette width is the average of the silhouette widths of the clusters. When clustering, we find the average silhouette width output from PAM for 1,2,...,k clusters. We take the number of clusters with the maximum average silhouette width as the number of clusters chosen by the biweight or Pearson's metric. We use the adjusted Rand, a measure of the clustering as compared to the known structure, to evaluate how well the biweight and Pearson's metric's re-cluster the simulated clusters. We explain the adjusted Rand in the subsection below. We want to see to what degree our best clustering, as chosen by maximizing the average silhouette width, matches our predicted groups.

4.3.1 Silhouette Width

The silhouette measures how well a point fits in its own cluster against how it fits in another cluster. For a gene, j, let a_j be the average dissimilarity of gene j with the other elements of its cluster, and let b_j be the minimum average dissimilarity of gene j with the members of all other clusters. To further explain b_j, let $q_1 ... q_n$ represent the points in cluster Q, a cluster of which j is not a member. Find the average dissimilarity between gene j and all $q_1 ... q_n$ in cluster Q. Continue to find the average dissimilarity between j and all genes within a cluster for all other clusters of which j is not a member.

Find the closest value of all the average dissimilarities found between clusters of which j is not a member; call it b_j. Hence, b_j is a quantification of how close the nearest cluster is, whereas a_j is a quantification of how close the elements of element j's cluster are to gene j. The silhouette of gene j can be any value -1 to 1 and will be given as:

$$S_j = \frac{b_j - aj}{\max(a_j, bj)} \tag{4.1}$$

There are three ways a gene can be "classified." If the dissimilarities within the cluster are less than the dissimilarity between clusters, the genes are "well-classified." If these dissimilarities are about the same, we do not know whether gene j should be in its current cluster or another. If the dissimilarity within the cluster is greater than that between clusters, then we have a "misclassified" gene. The values range from -1 to 1, i.e., from misclassified to well-classified, respectively.

The silhouette process is quite simple and intuitive, and finding the average of the silhouettes tends to be a good predictor of how well the genes are clustered. [14]

4.3.2 Adjusted Rand

The adjusted Rand is a cluster-validation measure described in detail by Yeung and Ruzzo [29]. The adjusted Rand is used to judge the clustering done by PAM - in our case - to external criteria, i.e., the simulated clusters we first created. To understand the adjusted Rand, first we consider the Rand index:

"Given a set of n objects $S = (O_1, ..., O_n)$, suppose $U = (u_1, ..., u_R)$ and $V = (v_1, ...v_C)$ represent two different partitions of the objects in S such that $\cup_{i=1}^{R} u_i = S = \cup_{j=1}^{C} v_j$ and $u_i \cap u_{i'} = \varnothing = v_j \cap v_{j'}$ for $1 \leq i \neq i^{'} \leq R$ and $1 \leq j \neq j^{'} \leq C$. Suppose that U is the external criterion and V is a clustering result. Let a to be the number of pairs of objects that are placed in the same class as U and in the same cluster in V, b to be the number of pairs of objects in the same class in U,but not in the same cluster in V, c be the number of pairs of objects in the same cluster in V but not in the same class in U, d be the number of pairs of objects in different classes and different clusters in both partitions" [29]. The Rand index is:

$$\frac{a + d}{a + b + c + d} \tag{4.2}$$

The Rand index will output a value between 0 and 1, and when the two partitions agree perfectly, the Rand index ouput is 1.

We use the adjusted Rand index instead of the Rand index, "because the expected value of the Rand index of two random partitions does not take a constant value" [29]. "The adjusted Rand index assumes the generalized

Table 4.1: Notation for the contingency table for comparing two partitions

Class	v_1	v_2	...	v_{1C}	*Sums*
u_1	n_{11}	n_{12}	...	n_{1C}	$n_{1.}$
u_2	n_{21}	n_{22}	...	n_{2C}	$n_{2.}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
u_R	n_{R1}	n_{R2}	...	n_{RC}	$n_{R.}$
Sums	$n_{.1}$	$n_{.2}$	...	$n_{.C}$	$n_{..} = n$

hypergeometric distribution as the model of randomness, i.e., the U and V partitions are picked at random such that the number of objects in the classes and clusters are fixed. Let n_{ij} be the number of objects that are in both class u_i and v_j. Let $n_{i.}$ and $n_{.j}$ be the number of objects in class u_i and v_j, respectively" [29]. The notations are illustrated in Table 4.1 above.

"The general form of an index with a constant expected value is $\frac{\text{index}-\text{expected index}}{\text{maximum index}-\text{expected index}}$, which is bounded above by, and is 0 when the index equals its expected value" [29].

Under the generalized hypergeometric model, it can be shown that the adjusted Rand index is:

$$\frac{\sum_{ij}\binom{n_{ij}}{2} - [\sum i\binom{n_{i.}}{2}\sum j\binom{n_{.j}}{2}]/\binom{n}{2}}{\frac{1}{2}[\sum_i\binom{n_{i.}}{2} + \sum j\binom{n_{.j}}{2}] - [\sum i\binom{n_{i.}}{2}\sum j\binom{n_{.j}}{2}]/\binom{n}{2}} \tag{4.3}$$

We use the form from equation 4.3 as the Rand index in our results. From here forward, the adjusted Rand values will be referred to as simply the Rand output or value.

Chapter 5

Initial Results of the Clusterings

5.1 Results when Simulating with the Absolute Euclidean and Euclidean Metrics

In both the large and small simulations, we get lower Rand values when the biweight metric clusters the data simulated by the Euclidean and absolute Euclidean metrics than when Pearson's metric clusters the same data. (See Tables 8.1 and 8.2) However, just because we get a lower Rand for biweight, we cannot say that clustering with Pearson's metric creates better clusters. In fact, we should expect that Pearson's metric would cluster data simulated by the Euclidean and absolute Euclidean metrics better, since Pearson's metric is a function of the Euclidean metric.

5.2 Results when Simulating with Spearman's Metric

For the data simulated by Spearman's metric, biweight and Pearson's perform similarly, with biweight displaying slightly higher Rand values (See Tables 8.1 and 8.2). Pearson's metric and the biweight metric both reclustered the simulated clustered with an average Rand of nearly 1 in the small simulations (See Table 8.2). Since Spearman's correlation is considered more resistant to outliers than Pearson's correlation, we expect Spearman's metric to not cluster together as many gene pairings with outliers as a non-resistant metric, like the Euclidean metric. The juxtaposition between Spearman's results and the Euclidean and absolute Euclidean results informed us that outliers may play an important role in how the Pearson's and biweight metrics re-cluster simulated data, since an important difference between Spearman's metric and the Euclidean metric (or absolute Euclidean metric) is the Spearman's metric's resistance to outliers.

5.3 Results when Simulating with the Biweight and Pearson's Metrics

We get a Rand of one or nearly one for all of the biweight metrics clusterings of the data the biweight metric simulated. (See Tables 8.2 and 8.1). Likewise, Pearson's metric clusters the data that Pearson's metric simulates with a Rand value of one or approximately one. We should expect the biweight metric to estimate high Rand values when re-clustering simulated clusters as defined by the biweight metric since biweight determined the distances between the genes to create the clusters in the first place. However, it is interesting that Pearson's also re-clusters the simulated clusters as created by biweight with a Rand of one. We hypothesize that since biweight is a metric that is resistant to outliers, in the initial clustering of the data, very few outliers within gene pairings will be clustered together. Hence, both Pearson's correlation and biweight correlation will either find gene pairings to be highly correlated, or not. Again, our results suggest that outliers play a key role in the clustering of microarray data.

5.4 Interpreting our Initial Results

From our results, we were unable to derive any consensus about which distance metric clusters simulated data with a higher Rand. We hypothesize that we were unable to derive a consensus about whether the biweight metric re-cluster simulated clusters into their original partitions better than Pearson's metric, because the specific distance metric used to simulate the data is fundamental to how the groups will be perceived by the distance metric used to cluster the data. For example, if Pearson's metric creates the simulated clusters, Pearson's metric will output a higher Rand value than the biweight metric when clustering the data. However, just because Pearson's metric clusters such that it outputs a higher Rand value, this does not imply that Pearson's metric is "right" in its designated clustering. It is hard to say what a "best" cluster is, since our initial simulations could have been poorly generated.

We hypothesize that the biweight metric may have not clustered the data according to their simulated partitions in some cases since the biweight metric is more resistant to outliers than Pearson's metric. In turn, the biweight metric may have clustered gene pairings differently than Pearson's metric because the biweight metrics measures closeness differently; specifically, the biweight metric gives different values to the distance between gene pairings than Pearson's metric when there are outliers between gene pairings.

5.5 A New Approach to Achieve our Goal

We found some interesting cases when examining the disparities between average Rand values when biweight and Pearson's metric re-cluster simulated data. Specifically, there were four cases when simulated clusters were re-clustered with a Rand of one by Pearson's metric, but with a lower Rand when biweight was the metric. Hence, in these cases, Pearson's metric clusters the data "perfectly" based on what we defined our initial groupings to be. Only two distance metrics, the Euclidean and absolute Euclidean metrics, simulated the four cases where Pearson's clustered with a Rand of one. The fact that clustering with Pearson's metric outputs a higher Rand than clustering with the biweight metric goes against our initial hypothesis that the biweight metric will cluster our simulated data into their "correct" groupings. Thus, we wanted to closer examine these instances.

We analyze the instances when Pearson's metric clusters the simulated data such that it outputs a Rand of one and the biweight metric clusters the simulated data such that it outputs a Rand of less than one. We call these instances "perfect instances." We hope to show that the differences in clusterings made by the biweight metric are due to outliers creating significant differences in the estimates of closeness between two genes made by the biweight metric and Pearson's metric.

5.6 Examining Differences in Clusterings

In order to investigate the differences in the "perfect instances" of the clusterings by the biweight metric and Pearson's metric, we decided to look at how many extreme outliers were clustered differently by the biweight and Pearson's metrics. Hence, we need to define an extreme outlier. By measuring the pairwise distances of all 3360 genes in a robust way, we can define a robust cutoff value, beyond which a pair of genes is considered extreme. Our results in the next chapter are an investigation into how extreme outliers affect the output of "perfect instances." In the sections below, we define an extreme outlier by finding robust measures for all pairwise distances and a cutoff value, beyond which the robust measure of a pairwise distance is an extreme outlier.

5.7 A Robust Measure of Pairwise Distances

Our pairwise distances are calculated using the Mahalanobis Squared Distance (or MSD) measurement:

$$d^2_{MSD,i}(x,y) = (\begin{pmatrix} x_i \\ y_i \end{pmatrix} - \begin{pmatrix} \overline{X}_{MCD} \\ \overline{Y}_{MCD} \end{pmatrix})^T S^{-1}_{MCD}(\begin{pmatrix} x_i \\ y_i \end{pmatrix} - \begin{pmatrix} \overline{X}_{MCD} \\ \overline{Y}_{MCD} \end{pmatrix}) \tag{5.1}$$

where MCD = $(\overline{X^*}_J, S^*_J)$ where J = [set of h points: $|S^*_J| \leq |S^*_K| \quad \forall$ sets K s.t. $\#|K| = h$] where $\#|w|$defines the number of elements in set w. That is, the MCD is the mean vector and scatter matrix of the subset of points of size h with the smallest covariance determinant.

$$\overline{X}^*_{MCD} = 1/h \sum_{i \in J} x_i \tag{5.2}$$

$$S^*_{MCD} = 1/h \sum_{i \in J} (x_i - \overline{X}^*_J)(x_i - \overline{X}^*_J)^t \tag{5.3}$$

$h = \lfloor \frac{(n+p+1)}{2} \rfloor$, n=17, i.e., the number of arrays in our data, and p=2. The value h is the highest possible breakdown point for the MCD, i.e., it is the minimum number of points which must not be outlying.

5.8 Finding Cutoffs

Once we know the $d^2_{MSD,i}$ for each pairwise distance, we need a mechanism for evaluating whether the distance is sufficiently outlying with respect to the relationship between the two genes. MSD distances with MCD shape and location parameters are known to be robust with an F-distribution when the data are normally distributed [9].

Hence, we want to find a cutoff value using an F-distribution to choose which points are outliers. We use the similarity equation:

$$\frac{c(m-p+1)}{pm} d^2_{S^*}(X_i, \overline{X^*}) \backsim F_{p,m-p+1} \tag{5.4}$$

to find $d^2_{S^*}(X_i, \overline{X^*})$, our cutoff value.

To find $d^2_{S^*}(X_i, \overline{X^*})$ from equation 5.4, we need estimates of c and m. To find an estimate of c, $\hat{c}$, and an estimate of m, $\hat{m}$, we use the set of equations below. $\hat{c}$ is solved for in equation 5.5, and we use equations 5.6-5.17 to solve for $\hat{m}$.

$$\begin{aligned}
\hat{c} &= \frac{P(\chi^2_{p+2} < \chi^2_{p,h/n})}{h/n} && (5.5)\\
\alpha &= \frac{n-h}{n} && (5.6)\\
1-\alpha &= P(\chi^2_p \leq q_\alpha) && (5.7)\\
c_\alpha &= \frac{1-\alpha}{P(\chi^2_p \leq q_\alpha)} && (5.8)\\
c_2 &= \frac{-P(\chi^2_{p+2} \leq q_\alpha)}{2} && (5.9)\\
c_3 &= \frac{-P(\chi^2_{p+4} \leq q_\alpha)}{2} && (5.10)\\
c_4 &= 3 * c_3 && (5.11)\\
b_1 &= \frac{c_\alpha(c_3 - c_4)}{1-\alpha} && (5.12)\\
b_2 &= .5 + \frac{c_\alpha}{1-\alpha}(c_3 - \frac{q_\alpha}{p}(c_2 + \frac{1-\alpha}{2})) && (5.13)\\
v_1 &= (1-\alpha)b_1^2(\alpha(\frac{c_\alpha q_\alpha}{p} - 1)^2 - 1) - 2c_3c_\alpha^2(3(b_1 - pb_2)^2 + \\
&\quad (p+2)b_2(2b_1 - pb_2)) && (5.14)\\
v_2 &= n(b_1(b_1 - pb_2)(1-\alpha))^2 c_\alpha^2 && (5.15)\\
v &= \frac{v_1}{v_2} && (5.16)\\
\hat{m} &= \frac{2}{c_\alpha^2} && (5.17)
\end{aligned}$$

Chapter 6

Results from Our Analysis of "Perfect Instances"

Within each of the clusters created by both Pearson's and biweight, we found the number of extreme outliers. The extreme outliers were tabulated to measure any effects outliers might have on the clustering. There were no systematic differences in the number of extreme outliers clustered together by Pearson's and biweight. (See Table 8.3). However, just because there were no systematic differences in the number of extreme outliers, our results are not moot. In fact, our results are just the opposite.

There were multiple types of outliers. To a non-resistant metric, some outliers make a gene pairing seem uncorrelated. But, after removing the outlier, the non-resistant metric finds the genes to be correlated. Biweight will cluster the gene pairing with the extreme outlier, since biweight estimates a high correlation with or without the outlier. On the other hand, some outliers make two genes seem correlated. But, after removing the outlier, the non-resistant metric find the genes to not be correlated. Biweight will not cluster the gene pairing with the extreme outlier since biweight estimates a low correlation with or without the outlier. We also found a few isolated instances of multiple extreme outliers within the gene pairings in "perfect instances." The multiple types of extreme outlier pairs are broken into four categories; the four categories are displayed by Figures 8.1-8.4.

We consider a "robust" cluster to be a cluster that would be clustered the same with the removal of extreme outliers from gene pairings as without the removal. Our results show that in the instances when the biweight metric clusters extreme outlier pairs differently than Pearson's metric, biweight is creating more "robust" clusters. Since there are imperfections, such as human error, within many steps within the microarray data collection process, we expect there to be outliers in our data. Hence, it is an advantage of the biweight metric that it creates more "robust" clusters than Pearson's metric.

6.1 Analyzing the Genes Pearson's Clusters Together

Table 8.3 displays the number of pairwise differences for four "perfect instances." Of the nineteen small simulations (See Table 8.2), four were "perfect instances." Only two distance metrics, the Euclidean and absolute Euclidean metrics, simulated these "perfect instances." In Table 8.3, we break up the four "perfect instances" into two categories: gene parings that look like Figure 8.1 and gene pairings that look like Figure 8.2

First, we consider the case where an outlier makes two genes seem correlated to a non-resistance metric, but not correlated to a resistant metric. Visually, we can see an example of an extreme outlier making two genes seem correlated in a non-resistant metric, in Figures 8.1 and 8.2. Pearson's clusters together the two genes displayed in Figures 8.1 and 8.2, whereas biweight does not. Below, we quantitatively explain the biweight and Pearson's metric's estimates of correlation between the genes in Figures 8.1 and 8.2 to exemplify the resistance of the biweight metric to outliers.

The cases we consider in this section, visually displayed by Figures 8.1 and 8.2 refer to Graph 1 and Graph 2 in Table 8.4, respectively. There are seventy-five instances Pearson's clusters the pairwise genes together when biweight does not such that the graph looks like Figure 8.1, and there are twenty instances where the graph looks like Figure 8.2.

The pairwise distance the biweight metric estimates for the genes in Figure 8.1 is -0.046. The pairwise distance Pearson's metric estimates for the genes in Figure 8.2 is -0.89. After removing the extreme outlier from the two genes, the pairwise distance the biweight metric estimates for the genes in Figure 8.1 is .046, and the pairwise distance Pearson's metric estimates for the genes in Figure 8.2 is 0.045. The removal of the outlier changes the Pearson's correlation value by 0.845. Our results display that when clustering microarray data, there are some cases where Pearson's metric will cluster together two genes the biweight metric will not cluster together because the genes share an extreme outlier.

We now consider the case of when the graph looks like Figure 8.2. The biweight metric estimates a correlation of -.09 for the genes in Figure 8.1. Pearson's metric estimates a correlation of -0.67 for the gens in Figure 8.1. After removing the extreme outlier from the two genes, biweight estimates a correlation of -0.06 for the genes in Figure 8.1. Pearson's metric estimates a correlation of 0-.15 for the genes in Figure 8.1. The removal of the outlier changes the Pearson's correlation by 0.52. Hence, even though the graph may look different from the Figure 8.1, we still find that Pearson's will cluster together genes even though Pearson's does not find a correlation with the removal of an extreme outlier.

6.2 Analyzing the Genes Biweight Clusters Together

The cases we consider in this section, visually displayed by Figures 8.3 and 8.4 refer to Figure 3 and Figure 4 in Table 8.5, respectively. Notice, of the 109 instances the biweight metric clusters the pairwise genes together when Pearson's does not, there are only four instances where the graph looks like Figure 8.4. We found so few cases, because Figure 8.4 exemplifies the special case when two extreme outliers are within a gene pairing.

The biweight metric estimates a value of -0.646 for the genes in Figure 8.3. Pearson's estimatesa correlation of -0.08 for the genes in Figure 8.3. After removing the extreme outlier from the two genes, biweight estimates a correlation of -0.63 for the genes in Figure 8.3. Pearson's correlation estimates a correlation of -.067 for the genes in Figure 8.3. The removal of the outlier changes the Pearson's metric value by 0.55. Our results display that when clustering microarray data, there are some cases where Pearson's metric will not cluster together two genes biweight will cluster together because the genes share an extreme outlier.

We now consider the four instances of when the graph looks like 8.4. The biweight metric estimates a correlation of -0.30 for the genes in Figure 8.4 . Pearson's metric estimates a correlation of -0.10 for the genes in Figure 8.4. After removing the two extreme outliers from the two genes, the biweight metric estimates a correlation of -0.28 for the genes in Figure 8.4. After the removal of two outliers, Pearson's estimates a correlation of -0.32 for the genes in Figure 8.4. The removal of the two extreme outliers changes the Pearson's estimate by 0.22. Hence, even though the graph may look different from the Figure 8.4, we still find that Pearson's will not cluster together genes even though Pearson's does not find a correlation (similar to that biweight finds) with the removal of two extreme outliers.

Chapter 7

Conclusion

One goal of microarray analysis is to find genes that have similar expression patterns. The adjusted Rand is good at displaying the differences between how well a clustering algorithm performs based on an external criterion. However, a high adjusted Rand does not imply a “better” clustering method because the external criterion may not be objectively created.

We examined “perfect instances” because, in these cases, the Pearson’s metric supposedly clustered simulated groups into the appropriate clusters, whereas the biweight metric did not. However, we showed in the results that the different clustering choices that the biweight metric makes in these “perfect instances” are often due to its resistance to outliers. Hence, we believe that biweight’s clustering is actually better. However, we have not shown that biweight creates more valid clusters based on some objective truth. In future research, we would like our simulated clusters to define some objective truth about gene expression patterns.

Our results are evidence that non-robust metrics can often cluster genes that are not correlated; the clustering of non-correlated genes is dangerous to scientists making conclusions about trends in expression patterns. Hence, we believe our results are not only telling of the advantages of using a biweight metric in the clustering process, but they are also telling of the disadvantages to using a non-robust metric in the clustering process.

In our future research, we hope to further analyze the advantages of robust metrics in clustering microarray data. A separate research idea is to explore the advantages of tight clustering through resampling [24]. We can use tight clustering methods to find a best cluster, rather than forcing our data into k clusters. A best cluster may tell us all we need, especially if we can design the clustering method to look for groups of certain types of expression patterns.

Chapter 8

Appendix: Tables and Figures

Metric Used in Simulations	# Sims	Clustered with Pearson's	Clustered with biweight
Euclidean	5	0.772	0.348
Absolute Euclidean	5	0.660	0.234
Spearman's	5	0.766	0.796
Biweight	5	0.628	0.969
Pearson's	5	0.977	0.784

Each entry in columns 3 and 4 represents the average adjusted Rand over five simulations.

Table 8.1: Large Simulation Results

Metric Used in Simulations	# Sims	Clustered with Pearson's	Clustered with biweight
Euclidean	5	0.528	0.233
Absolute Euclidean	5	0.661	0.234
Spearman's	5	0.97	0.980
Biweight	1	0.674	1
Pearson's	1	1	1

Each entry in columns 3 and 4 represents the average adjusted Rand over the number of simulations defined in column 2.

Table 8.2: Small Simulation Results

Perfect Instances:	Abs. Eucl. 1	Abs. Eucl. 2	Eucl. 1	Eucl. 2	Total
# Pairwise Differences	82	34	29	39	184
#B	38	23	16	32	109
#P	44	11	13	7	75

Each entry in # Pairwise Difference represents the number of gene pairs with extreme outliers clustered differently by Pearson's and biweight
Each entry in #B represents the number pairwise differences clustered together by the biweight metric
Each entry in #P represents the number of pairwise differences clustered together by Pearson's metric

Table 8.3: Results from Our Analysis of Extreme Outliers

Perfect Instances:	Abs. Eucl. 1	Abs. Eucl. 2	Eucl. 1	Eucl. 2	Total
Looks like Graph 1	31	10	9	5	55
Looks like Graph 2	13	1	4	2	20

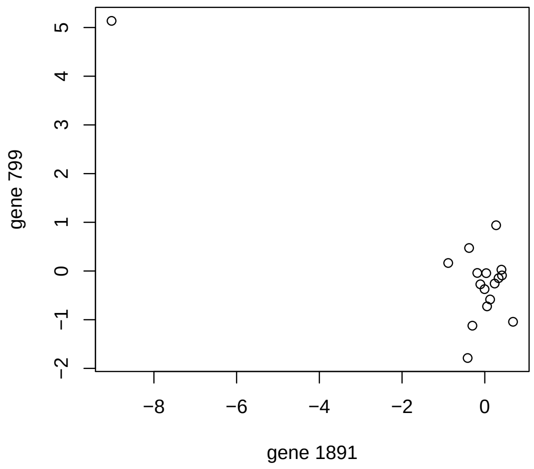
Each of the pairwise differences clustered together by Pearson's is similar to one of two graphs, Graph 8.1 or Graph 8.2.

Table 8.4: Pearson's Results from Our Analysis of Extreme Outliers

Perfect Instances:	Abs. Eucl. 1	Abs. Eucl. 2	Eucl. 1	Eucl. 2	Total
Looks like Graph 3	38	23	12	32	105
Looks like Graph 4	0	0	4	0	4

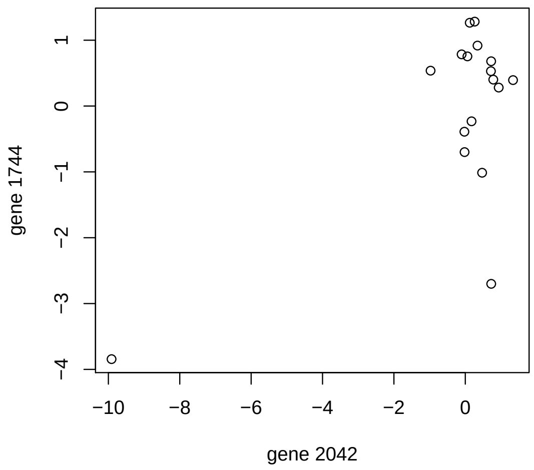
Each of the pairwise differences clustered together by biweight is similar to one of two graphs, Graph 8.3 or Graph 8.4.

Table 8.5: Biweight Results from Our Analysis of Extreme Outliers



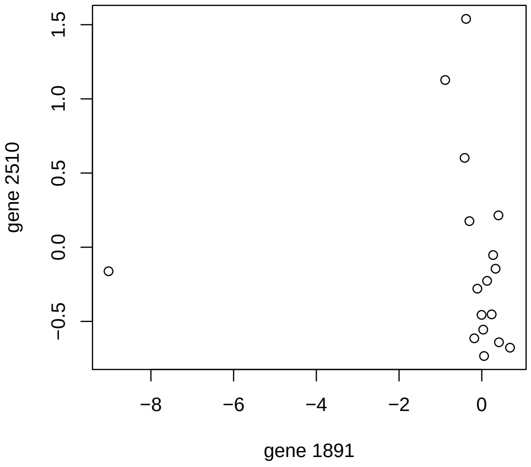
Each gene comes from our data set of 3360 genes

Figure 8.1: Graph 1: Pearson's Clusters Two Genes Together



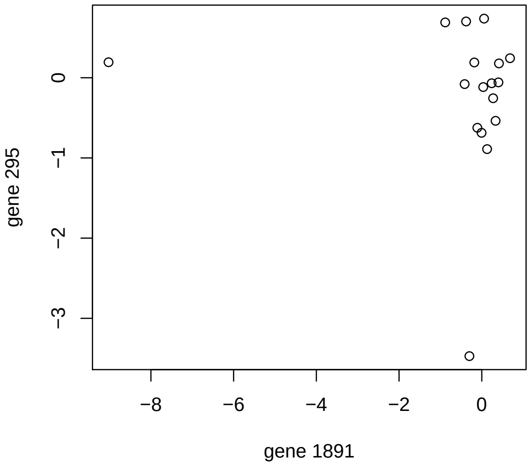
Each gene comes from our data set of 3360 genes

Figure 8.2: Graph 2: Pearson's Clusters Two Genes Together



Each gene comes from our data set of 3360 genes

Figure 8.3: Graph 3: Biweight Clusters Two Genes Together



Each gene comes from our data set of 3360 genes

Figure 8.4: Graph 4: Biweight Clusters Two Genes Together

Bibliography

[1] Butte, Atul. “The Use and Analysis of Microarray Data.” Nature Reviews Drug Discovery. 1.12 (2002): 951-960.

[2] Costa, Ivan G., Francisco de A.T. de Carvalho, and Marcilio C.P. de Souto. “Comparative Analysis of Clustering Methods for Gene Expression Time Course Data.” Genetics and Molecular Biology. 27.4 (2004): 623-631.

[3] Datta, Susamita, and Somnath Datta. “Comparisons and Validation of Statistical Clustering Techniques for Microarray Gene Expression Data.” Bioinformatics. 19.4 (2003): 459-466.

[4] D’Haeseleer, Patrik. “How Does Gene Expression Clustering Work?” Nature Biotechnology. 23.12 (2005): 1499-1501.

[5] Draghici, Sorin. Data Analysis Tools for DNA Microarrays. Boca Raton, FL: Chapman and Hall/CRC Press, 2003.

[6] Everitt, Brian S., Sabine Landau, and Morven Leese. Cluster Analysis. 3rd ed. London: Oxford University Press, 2001.

[7] Gentleman, Carey, Wolfgang Hubor, Vincent J. Carey, Rafael A. Irizarry, and Sandrine Dudoit. Bioinformatics and Computational Biology Solutions Using R and Bioconductor. New York: Springer Science and Business Media, Inc, 2005.

[8] Gibbons, Francis D., and Frederick P. Roth. “Judging the Quality of Gene Expression-Based Clustering Methods Using Gene Annotation.” Genome Research. 12 (2002): 1574-1581.

[9] Hardin, Johanna, and David Rocke. “The Distribution of Robust Distances.” Journal of Computational and Graphical Statistics. 14. 4 (2005): 1-19.

[10] Hardin, Johanna, and David Rocke. Outlier Detection in the Multiple Cluster Setting Using the Minimum Covariance Determinant Estimator. Computational Statistics & Data Analysis. 44.4 (2004): 625-638.

[11] Hardin, Johanna S., Aya Mitani, Leanne Hicks, and Brian VanKoten. “A Robust Measure of Correlation between Two Genes on a Microarray. BMC Bioinformatics. 8 (2007): 1-25.

[12] Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. The Elements of Satistical Learning: Data Mining, Inference, and Prediction. 3rd ed. Stanford University, CA: Springer, 2003.

[13] Hoaglin, David C., Frederick Mosteller, and John W. Tukey. “M-Estimators of Location: An Outline of the Theory.” Understanding Robust and Exploratory Data Analysis. New York: John Wiley & Sons, Inc., 2000. ch. 3. 339-403.

[14] Kaufman, Leonard, and Peter J. Rousseeuw. Finding Groups in Data. Brussels: John Wiley and Sons, Inc.,1990.

[15] Magalhaes, Tiago R., Jessica Palmer, Pavel Tomancak, and Katherine S. Pollard. “Integration of Controlled Vocabularies and Hopach Clustering of Drosophila Microarray Data.” Bioinformatics. 00.00 (2006): 1-4.

[16] Pan, Wei, Lin Jizhen, and Chap T Le. “Model-Based Cluster Analysis of Microarray Gene-Expression Data.” Genome Biology. 3.2 (2002): 465-469.

[17] Pinkel, Daniel. “Cancer Cells, Chemotherapy and Gene Clusters.” Nature Genetics. 24 (2000): 208-209.

[18] Protter, Murray H., and Charles B. Morrey. A First Course in Real Analysis: Second Addition. New York: Springer-Verlag, Inc., 1991.

[19] Quackenbush, John. “Computational Analysis of Microarray Data.” Nature Genetics. 2 (2001): 418-427.

[20] Quackenbush, John. “Microarrays: Guilt by Association.” Science. 302 (2003): 240.

[21] Ross, Douglas T., et al. “Systematic Variation in Gene Expression Patterns in Human Cancer Cell Lines.” Nature Genetics—. 24 (2000): 227-235.

[22] Salvador, Stan, and Phillip Chan. Determining the Number of Clusters/Segments in Hierarchical Clustering Segmentation Algorithms. Washington, D.C.:IEEE Computer Society, 2004.

[23] Thalamuthu, Anbupalam, Indranil Mukhopadhyay, Xiaojing Zheng, and George C. Tseng. "Evaluation and Comparison of Gene Clustering Methods in Microarray Analysis." Bioinformatics. 22.19 (2006): 2405-2412.

[24] Tseng, George C., and Wing H. Wong. "Tight Clustering: A Resampling-Based Approach for Identifying Stable and Tight Patterns in Data." Biometrics. 61 (2005): 10-16.

[25] Viks, Gat, R. Sharan, and R. Shamir. "Scoring Clustering Solutions by their Biological Relevance." Bioinformatics. 19.18 (2003): 2381-2389.

[26] Wilcox, Rand R. Inferences Based on a Skipped Correlation Coefficient. Journal of Applied Statistics. 31. 2 (2004): 131-143.

[27] Wilcox, Rand R., and Jan Muska. "Inferences about Correlations when there is Heteroscedasticity." Journal of Mathematical and Statistical Psychology. 54 (2001): 39-47.

[28] Wong Dorothy S.V., Frederick K. Wong, and Graham R. Wood. "A Multi-Stage Approach to Clustering and Imputation of Gene Expression Profiles. Bioinformatics. 23.8 (2007): 998-1005.

[29] Yeung, Ka Yee, and Walter L. Ruzzo. "Details of the Adjusted Rand Index and Clustering Algorithm: Supplement to Principal Component Analysis for Clustering Gene Expression Data." Bioinformatics. 17.9 (2001): 763-774.