University of California, Berkeley
U.C. Berkeley Division of Biostatistics Working Paper Series

Year 2002 *Paper* 107

A Method to Identify Significant Clusters in Gene Expression Data

Katherine S. Pollard* Mark J. van der Laan†

*Division of Biostatistics, School of Public Health, University of California, Berkeley, kpollard@gladstone.ucsf.edu

†Division of Biostatistics, School of Public Health, University of California, Berkeley, laan@berkeley.edu

http://biostats.bepress.com/ucbbiostat/paper107

A Method to Identify Significant Clusters in Gene Expression Data

Katherine S. Pollard and Mark J. van der Laan

Abstract

Clustering algorithms have been widely applied to gene expression data. For both hierarchical and partitioning clustering algorithms, selecting the number of significant clusters is an important problem and many methods have been proposed. Existing methods for selecting the number of clusters tend to find only the global patterns in the data (e.g.: the over and under expressed genes). We have noted the need for a better method in the gene expression context, where small, biologically meaningful clusters can be difficult to identify. In this paper, we define a new criteria, Mean Split Silhouette (MSS), which is a measure of cluster heterogeneity. We propose to choose the number of clusters as the minimizer of MSS. In this way, the number of significant clusters is defined as that which produces the most homogeneous clusters. The power of this method compared to existing methods is demonstrated on simulated microarray data. The minimum MSS method is an example of a general approach that can be applied to any clustering routine with any global criteria.

1 Motivation

Gene expression studies are swiftly becoming a very significant and prevalent tool in biomedical research. The microarray and gene chip technologies allow researchers to monitor the expression of thousands of genes simultaneously. An important goal with large-scale gene expression studies is to find biologically important subsets of genes and samples. Clustering algorithms have been widely applied to this problem. These can be classified into partitioning and hierarchical clustering algorithms. Examples of hierarchical algorithms include agglomerative clustering (as implemented in the Cluster program by Eisen *et al.* [3]) and HOPACH [14]. Partitioning algorithms include K-Means, Self-Organizing Maps [12], and Partitioning Around Medoids [6]. With both types of algorithms, it is necessary to select the number of significant clusters. In the hierarchical tree context this corresponds with choosing the level of the tree at which the clusters are still significant. Methods for determining the number of clusters are reviewed and compared by Milligan and Cooper [7] and by Fridlyand and Dudoit [5], who note that none of the existing methods are satisfactory for gene expression data analysis. We have also noted the need for a better method in the gene expression context, where small, biologically meaningful clusters can be difficult to identify. In particular, existing methods tend to identify only the global structure in the data, for example, the over and under expressed genes. In this paper, we present a new method for selecting the significant clusters.

We begin by describing the context and type of data that motivated our method in Section 2. In Section 3, we first present a general method for identifying significant clusters and then illustrate the method with a specific criteria function, Mean Split Silhouette (MSS). The power of this approach compared to existing methods is demonstrated on simulated microarray data in Section 4.

2 Background

A typical gene expression experiment results in an observed data matrix X whose columns are n copies of a p-dimensional vector of gene expression measurements, where n is the number of observations and p is the number of genes. For microarrays, each measurement is typically a ratio, calculated from the intensities of two flourescently labeled mRNA (or cDNA) samples cohybridized to arrays spotted with known cDNA sequences. Gene chips produce similar data, except each element is a quantitative expression level rather than a ratio. In both cases, the genes are a set of p elements $\mathbf{x}_j$, $j \in \{1, \ldots, p\}$, where each element $\mathbf{x}_j$ is an n dimensional vector $(x_{1j}, \ldots, x_{nj})^T$. Similarly, the n samples are each a p dimensional vector. The methods we present can be used with either microarray or gene chip data (or indeed any high dimensional data), but for simplicity we will assume that the measurements are ratios.

Given data from such an experiment, researchers are often interested in identifying groups of differentially expressed genes which are *significantly correlated with each other*, since such genes might be part of the same causal mechanism or pathway.

In addition to identifying interesting clusters of genes, researchers often want to find subgroups of samples (e.g.: patients) which share a common gene expression profile. Thus, the data is usually first screened to eliminate certain genes, such as those showing no difference in expression, from the subset. Then, the genes and/or the samples are clustered. We will assume in our explanation of the method that the set of elements to be clustered is the genes. In the simulations (Section 4), we illustrate both gene and sample clustering. We have proposed a statistical framework for simultaneous clustering of both genes and samples [8].

All clustering algorithms are either implicitly or explicitly functions of a dissimilarity matrix which measures the distance between every pair of elements. Let $d(\mathbf{x}_j, \mathbf{x}_{j'})$ denote the dissimilarity between elements j and j' and let $\mathbf{D}$ be the $p \times p$ symmetric matrix of dissimilarities. Typical choices of dissimilarity include Euclidean distance, 1 minus correlation, 1 minus absolute correlation and 1 minus cosine-angle. For example, the cosine-angle distance between two vectors was used in [3] to cluster genes based on gene expression data across a variety of cell lines.

Partitioning methods generally require that the user specify the number of clusters, whereas hierarchical methods produce a tree of clusters in which each level is part of a nested series of clustering results with sequentially more clusters as one moves from top to bottom. With both types of methods, identifying a main clustering result corresponds with choosing the number of clusters. Choosing the number of clusters in a data analysis is equivalent to estimating the true number of clusters, which is a parameter of the true data generating distribution defined by the clustering method and the criteria for selecting the number of clusters. Different criteria for selecting the number of clusters may estimate different parameters, so that it is important to perform simulations in order to understand how a particular criteria works and to decide when it will be useful.

Methods for selecting the number of significant clusters include direct methods and resampling methods. Direct methods consist of optimizing a criteria, such as functions of the within and between cluster sums of squares [7], occurrences of phase transitions in simulated annealing [9], likelihood ratios [10], and average silhouette [6]. The method of maximizing average silhouette has the advantage of being able to be used with any clustering routine and any distance metric. A disadvantage of average silhouette is that, like many criteria functions for selecting the number of clusters, average silhouette measures the global structure only. We discuss this problem in more detail below. Resampling methods take a different approach, testing for significant evidence against a specific null hypothesis (e.g.: uniformity or unimodality) corresponding to no clusters. Examples of resampling methods that have been used with gene expression data are the gap statistic [11], the weighted average discrepant pairs (WADP) method [1], and Clest [5]. We have also proposed resampling methods based on the sensitivity and specificity of clusters over bootstrap samples [13] [2]. These resampling methods are computationally much more difficult than direct methods since they involve the creation and evaluation of many data sets. The method we present in this paper is a direct method, so we restrict our evaluation of it to a comparison with other direct methods.

3 Method

We have found some cases, in both real and simulated gene expression data, where existing clustering routines and methods for selecting the number of clusters fail to find the main clusters. The problem of finding relatively small clusters in the presence of one or more larger clusters is particularly hard. Another challenging problem arises when the clusters are not equally distant from each other, but rather form nested clusters within clusters as illustrated in Figure 1. This type of data structure arises frequently in gene expression data. In this context, it is frequently the finer structure that is of interest biologically. For example, the global structure might consist of two clusters (e.g.: over and under expressed genes), but the biologist may be interested in a particular, small cluster within the under expressed genes. This small cluster might only be apparent after "diving into" the context of a larger, more inclusive cluster, just as the details of neighborhoods are only visible from an airplane as one descends into the city. Many methods for identifying the number of clusters in a data set find only the global structure. Inspired by this lack of performance, we present a method for selecting the number of clusters in a data set which can be applied with both partitioning and hierarchical clustering algorithms.

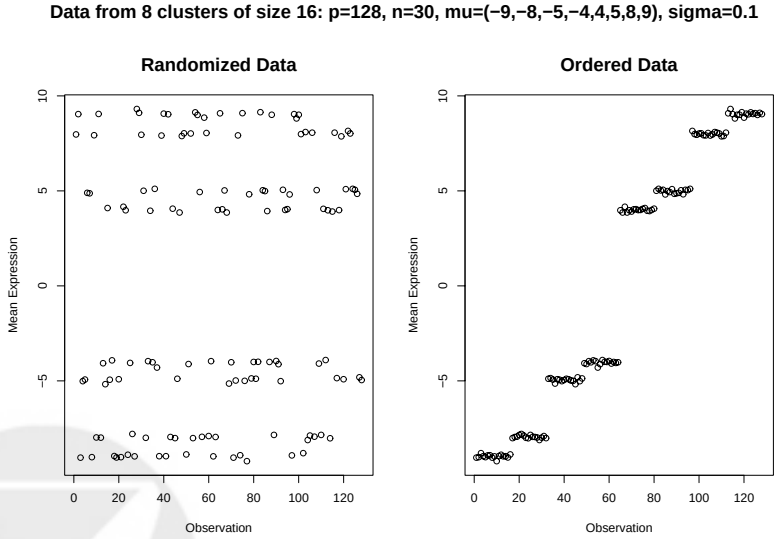

Figure 1: This example of a data set with nested clusters illustrates the difference between global and finer clustering structure. Each data point ($p = 128$) was generated independently from an $n = 30$ dimensional Normal distribution with one of eight means and a shared standard error $\sigma = 0.1$. The left panel is a plot of the data in random order. The right panel is a plot of the same data ordered by the mean of the Normal distribution from which it was generated. At the global level, there are two clusters consisting of positive versus negative means. The gap between these two sets is larger than any other gap in the data. However, if either set is considered independently, it is

easy to see two groups within it separated by a smaller gap. Similarly, there are two groups within each of these groups, so that at the finest level it is possible to distinguish eight clusters.

3.1 General Method

Consider a series of proposed clustering results and a global criteria function. With a partitioning algorithm, these may consist of applying the clustering routine with $k = 2, 3, \ldots, K$ clusters where K is a user-specified upper bound on the number of clusters. With a hierarchical algorithm, the series may correspond to levels of the tree. In either case, evaluate each proposed result separately using the following method. For each such result, apply the clustering routine independently to the elements in each of the clusters (ignoring elements in other clusters). Then evaluate the criteria function on each of these clustering results to obtain a measure of cluster heterogeneity for each cluster. Average this measure over clusters. Repeat the procedure for each of the proposed clustering results in the series. The minimum indicates the clustering result with most homogeneous clusters. The key idea behind the method is to evaluate how well the elements in a cluster belong together by diving into each cluster and applying the clustering algorithm and criteria function to the elements in that cluster alone, ignoring the other clusters. This approach can be applied with any clustering routine and any criteria function.

3.2 Mean Split Silhouette (MSS)

We present a particular application of this method called Mean Split Silhouette (MSS), which uses average silhouette as the criteria. Since silhouettes can be calculated with any clustering algorithm and any distance metric, MSS can be used to determine the number of clusters with all partitioning and hierarchical clustering algorithms. Suppose we are clustering genes.

Silhouette: Given a clustering, the silhouette for a given gene is calculated as follows [6]. For each gene j, calculate a_j which is the average dissimilarity of gene j with other elements of its cluster:

$$a_j = \operatorname{avg} d(\mathbf{x}_j \ , \mathbf{x}_{j'}), j' \in \{i : l_1(\mathbf{x}_i, M) = l_1(\mathbf{x}_j \ , M)\}.$$

For each gene j and each cluster k to which it does not belong (that is, $k \neq l_1(\mathbf{x}_j \ , M)$), calculate b_{jk}, which is the average dissimilarity of gene j with the members of cluster k:

$$b_{jk} = \operatorname{avg} d(\mathbf{x}_j \ , \mathbf{x}_{j'}), j' \in \{i : l_1(\mathbf{x}_i, M) = k\}.$$

Let $b_j = \min_k b_{jk}$. The silhouette of gene j is defined by the formula:

$$S_j(\mathbf{M}) = \frac{b_j - a_j}{\max(a_j, b_j)}. \tag{1}$$

Note that the largest possible silhouette is 1, which occurs only if there is no dissimilarity within gene j's cluster (*i.e.*: $a_j = 0$). The other extreme is -1. Heuristically,

the silhouette measures how well matched an object is to the other objects in its own cluster versus how well matched it would be if it were moved to the next closest cluster.
Average Silhouette: The average silhouette over all elements has been used to evaluate and compare clustering results, including selecting the number of clusters k by maximizing average silhouette over a range of possible values for k [6]. It has been our experience, based on simulated and real gene expression data, that the average silhouette is actually a very good global measure of the strength of clustering results: see also [4] for a favorable performance of average silhouette relative to other validation functionals. As we have argued, however, it is important to go beyond global structure in the analysis of gene expression data. Average silhouette alone is not able to identify this finer structure, as illustrated in Section 4.
Mean Split Silhouette (MSS): Given a clustering result with k clusters, consider splitting each cluster into two or more clusters (the number of which can be determined, for example, by maximizing average silhouette). In the hierarchical tree context, this corresponds with computing the child clusters in the next level of the tree, while in the partitioning context it corresponds with treating the elements in each cluster as a new sample and partitioning them. In both cases, each element has a new silhouette after the split, which is computed relative to only those elements with which it shares a parent. We call the average of the these for each parent cluster the split silhouette $SS_i, i = 1, 2, \ldots, k$. The split silhouette is a measure of that cluster's heterogeneity (i.e.: it is low if the cluster is homogeneous and should not be split). We define MSS as the mean of the split silhouettes over the k clusters:

$$MSS(k) = \frac{1}{k} \sum_{i=1}^{k} SS_i. \tag{2}$$

Then, MSS is a measure of the average heterogeneity of the clusters in the clustering result.
Choosing the Number of Clusters: Given a series of clustering results, we propose to select the number of clusters by choosing the proposed result which minimizes MSS. In this way, we choose the number of significant clusters that produces (on average) the most homogeneous groups. One nice benefit of this approach is that it is possible to select one cluster (i.e.: no groups) without using a testing approach and defining a null distribution. Unlike most global criteria, MSS is defined for $k = 1$; it is in fact the usual average silhouette for the whole data set. If the data is homogeneous, the minimum MSS will occur at $k = 1$, as illustrated in Section 4.

4 Simulations

In order to illustrate the performance of the minimum MSS approach and compare it to existing methods for choosing the number of clusters, we performed analyses of simulated data.

4.1 Data

We designed a simulated data structure which is an idealization of the pattern of nested clusters that we have observed in real gene expression data. The data are generated from multivariate normal distributions. In each simulation, there are eight clusters. The clusters differ in their means, and these means are spaced such that pairs of clusters are closer to each other than to any other cluster and then pairs of these pairs are again closer to each other that to the other sets of pairs (see Figure 1). The elements are uncorrelated with a shared standard error. As the standard error increases, the distinction between the clusters becomes less clear. In fact, at the value of sigma for which normal confidence intervals for the means of the pairs of closest clusters begin to overlap, it is no longer possible to distinguish these clusters visually in a plot of the distance matrix (see Figure 2). We would expect a good method for selecting the number of clusters to shift from eight to four clusters at this point.

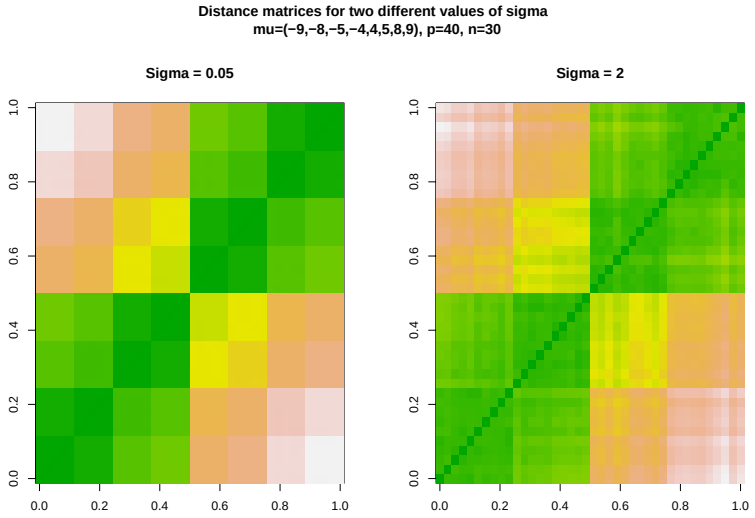

Figure 2: This example of a data set with nested clusters illustrates the effect of the noise level σ on the ability to identify the finer clustering structure. Each of the two Euclidean distance matrices is computed from a data set in which each data point ($p = 40$) is generated independently from an $n = 30$ dimensional Normal distribution with one of eight means and a shared standard error σ. The elements in each distance matrix are ordered by the mean of the Normal distribution from which they were generated. In the left panel, $\sigma = 0.05$ and it is easy to visually distinguish the eight clusters. In the right panel, $\sigma = 2$ and it is only possible to visually distinguish four clusters. In both plots, the pairwise distances are depicted on a color scale from green to white, with dark green denoting the smallest distance.

This general data structure is used to generate data sets for both gene and patient clustering examples. For gene clustering, we generate data sets with $p = 512$ genes

and $n = 30$ samples. We suppose that the data set has already been pre-screened to remove genes (frequently the majority) showing little difference in expression, leaving 512 genes of interest. The mean vector is $\mu = (-9, -8, -5, -4, 4, 5, 8, 9)$, which has the typical gap between over and under expressed genes that is observed after pre-screening. The clusters are of equal size so that each contains 64 genes. We consider a range of values for the standard error $\sigma \in \{0.05, 0.5, 0.95, 1.5, 2, 5\}$, and we expect to be able to distinguish only four clusters for $\sigma \in \{2, 5\}$.

For sample clustering, we generate data sets which resemble those we have found when following a method which we have proposed for simultaneously clustering genes and patients [8]. We suppose that the genes have already been clustered and an interesting cluster of over expressed genes was identified in which the patients differ in the amount they over express the subset of genes. We reduce the vector of gene expression measurements for each patient to the mean value over the genes in this subset. The resulting data set has $p = 1$ measurement for each of $n = 360$ patients. This large number of patients is plausible in the context where an interesting subset of genes has been pre-identified so that a larger sample can have expression of only these genes measured (possibly using technology other than microarrays). The mean vector is $\mu = (1, 2, 5, 6, 14, 15, 18, 19)$, representing 1 to 19-fold average over expression. The clusters are of equal size so that each contains 45 patients. We consider a range of values for the standard error of the mean $\sigma \in \{\approx 0, 0.01, 0.05, 0.15, 0.25, 0.5, 0.95, 0.99\}$, and we expect to be able to distinguish only four clusters for $\sigma \in \{0.5, 0.95, 0.99\}$. We look at $\sigma \approx 0$ in order to understand the clustering result for the true data generating distribution (i.e.: if we had observed the true distance matrix from the distribution in which each patient has the exact mean value).

For the gene clustering simulations, we use Euclidean distance, which is capable of capturing distinctions between clusters based on differences in their means. Euclidean distance could also be used in the patient clustering simulations. We choose to use the absolute value of the difference between the means in order to illustrate the use of a different distance metric.

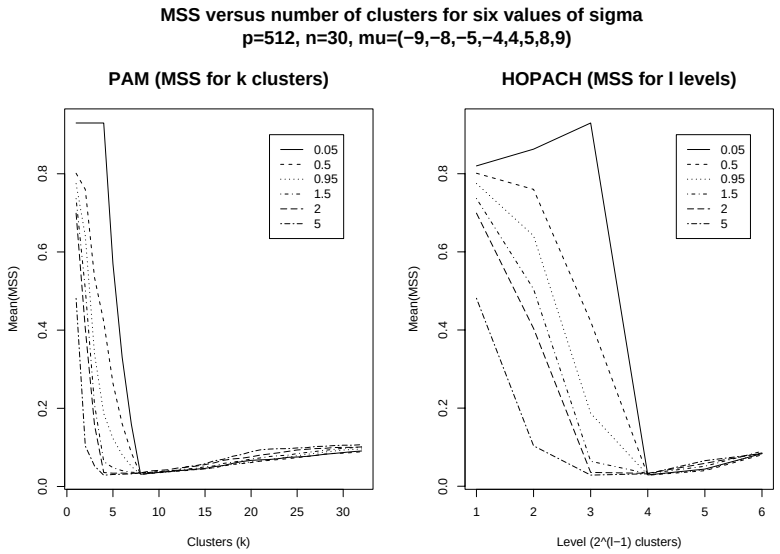

Figure 3: Gene Clustering Simulation - The mean value of MSS over ten independent data sets is plotted versus the number of clusters for a range of values of σ.

4.2 Clustering Routines

We clustered each of the simulated data sets with both a partitioning and a hierarchical method, each described briefly here.

Partitioning - PAM: The clustering procedure PAM [6] takes as input a dissimilarity matrix $\mathbf{D}$ and produces as output a set of cluster centers or "medoids". Let K be the number of clusters and let $\mathbf{M} = (M_1, \ldots, M_K)$ denote any size K collection of the n elements $\mathbf{x}_j$. Given $\mathbf{M}$, we can calculate the dissimilarity $d(\mathbf{x}_j\ , M_k)$ of each element and each member of $\mathbf{M}$. For each element $\mathbf{x}_j$, we denote the minimum and minimizer by $\min_{k=1,\ldots,K} d(\mathbf{x}_j\ , M_k) = d_1(\mathbf{x}_j\ , \mathbf{M})$ and $\min^{-1}_{k=1,\ldots,K} d(\mathbf{x}_j\ , M_k) = l_1(\mathbf{x}_j\ , \mathbf{M})$. PAM selects the medoids $\mathbf{M}^*$ by minimizing the sum of such distances $\mathbf{M}^* = \min^{-1}_{\mathcal{M}} \sum_j d_1(\mathbf{x}_j\ , M)$. Each medoid M^*_k identifies a cluster, defined as the elements which are closer to this medoid than to any other. This clustering is captured by a vector of labels $l(\mathbf{X}, \mathbf{M}^*) = (l_1(\mathbf{x}_1, \mathbf{M}^*), \ldots, l_1(\mathbf{x}_p, \mathbf{M}^*))$. Choosing the number of clusters with PAM corresponds with selecting the best possible value for K.

Hierarchical - HOPACH: Hierarchical Ordered Partitioning And Collapsing Hybrid (HOPACH) is a hybrid clustering method that applies a partitioning algorithm iteratively to produce a hierarchical tree of clusters [14]. At each node, a cluster is partitioned into two or more smaller clusters with an enforced ordering of the clusters. Collapsing steps can be applied at any level of the tree to unite similar clusters, correcting for errors made in the partitioning steps. A final ordered list is obtained by running down the tree completely. The ordering of elements at any level of the tree is deterministic and can be used to visualize the clustering structure in a colored plot of the

reordered data or distance matrix. The method combines the strengths of both divisive and agglomerative hierarchical clustering methods. For the purposes of this simulation, we used a version of HOPACH with PAM as the partitioning algorithm. After running the algorithm with a data-adaptive number of clusters at each node, we noticed that the data structure produced only binary splits. So, we restricted the number of clusters at each node to two for the analysis. Choosing the number of clusters with HOPACH corresponds with selecting a level of the tree to identify as the main clustering result.

4.3 Results

Clustering Genes: Figure 3 shows the mean value of MSS over ten independent simulated gene clustering data sets for each value of σ plotted versus the number of clusters. The variance of MSS is very low relative to the mean (especially near the correct number of clusters), so that the MSS plot for each individual data set resembles the mean value closely. The left panel illustrates the results from PAM clustering, where the horizontal axis is the number of clusters k. The right panel illustrates the results from HOPACH clustering, where the horizontal axis is the level of the tree l (with 2^{l-1} clusters). For both algorithms, the minimum MSS is at eight clusters for smaller values of σ and at four clusters for $\sigma > 1.5$, as expected.

Clustering Samples: Figure 4 shows the mean value of MSS over ten independent simulated patient clustering data sets for each value of σ plotted versus the number of clusters. The MSS plot for each individual data set resembles the mean value, since the variance of MSS is low relative to the mean (though slightly less so than in the genes simulation due to the lower dimension of the vectors being clustered). Again, PAM is illustrated on the left and HOPACH on the right. For both algorithms, the minimum MSS is at eight clusters for smaller values of σ and at four clusters for $\sigma \geq 0.5$.

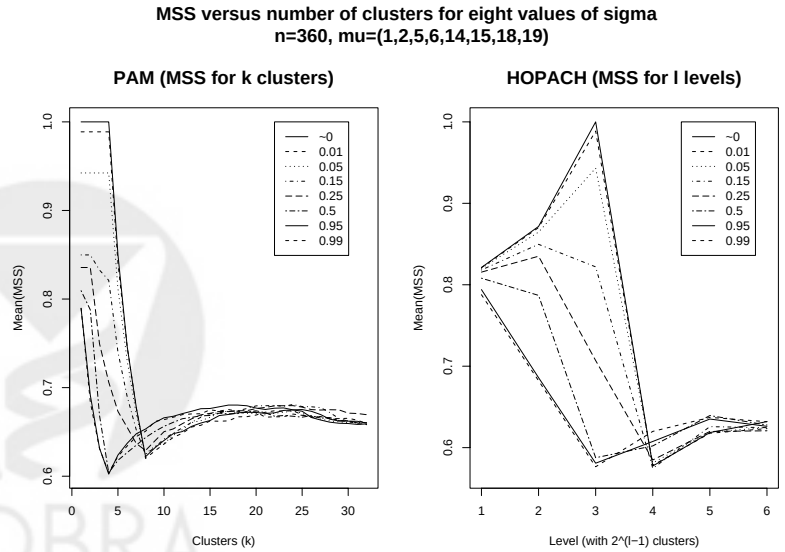

Figure 4: Patient Clustering Simulation - The mean value of MSS over ten independent data sets is plotted versus the number of clusters for a range of values of σ.

We also considered clustering a data set in which each patient is a p dimensional vector (p large), rather than the mean over a certain subset of genes. The results are similar, with MSS being even less variable due to the $n \times n$ patient distance matrix being extremely stable when computed over p genes.

One Cluster: Unlike most global criteria, MSS is defined for $k = 1$ cluster. It is in fact the usual average silhouette for the whole data set. Hence, it is possible to select one cluster (i.e.: no groups) using the minimum MSS method. In order to examine the performance of MSS with unimodal data (where we expect to find only one cluster), we simulated a data set with $n = 360$ observations from a univariate $N(0, 0.05)$ distribution, applied PAM and HOPACH with Euclidean distance and computed MSS. Figure 5 shows MSS versus the number of clusters for this data set, with PAM on the left and HOPACH on the right. For both algorithms, the minimum MSS is at one cluster. This result indicates that MSS can be used to directly select one cluster without using a testing approach and defining a null distribution.

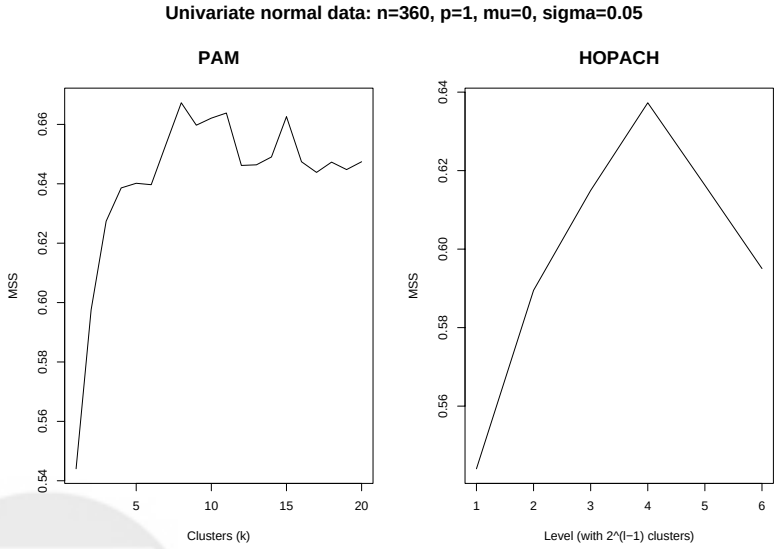

Figure 5: One Cluster Simulation - MSS is plotted versus the number of clusters.

Comparison with Other Methods: In order to compare the minimum MSS method to other methods for selecting the number of clusters, we implemented several methods from the literature on a simulated data set. We chose to compare MSS with all four of the direct methods in [5], which were also used in [11] and were among the best performing methods in [7]. Each of these methods is described briefly here. Let n be the number of elements to be clustered and p be the dimension of the elements. Define B_k and W_k as the $p \times p$ matrices of between and within k-clusters sums of squares and cross-products, and let $tr()$ denote the trace of a matrix. We seek an estimate $\hat{K}$ of the number of clusters.

1. Calinski & Harabasz (CH): $\hat{K} = argmax_{k \geq 2} CH_k$, where

$$CH_k = \frac{tr(B_k)/(k-1)}{tr(W_k)/(n-k)}.$$

2. Hartigan (H): $\hat{K} = argmin_{k \geq 1} H_k$ such that $H_k \leq 10$, where

$$H_k = \left(\frac{tr(W_k)}{tr(W_{k+1})} - 1 \right) (n-k-1).$$

3. Krzanowski & Lai (KL): $\hat{K} = argmax_{k \geq 2} KL_k$, where

$$d_k = (k-1)^{2/p} tr(W_{k-1}) - k^{2/p} tr(W_k),$$

$$KL_k = \frac{|d_k|}{|d_{k+1}|}.$$

4. Silhouette (Sil): $\hat{K} = argmax_{k \geq 2} sil_k$, where sil_k is the average silhouette over all elements and silhouette is defined in Eq.(1).

We used the simulated data models from the gene and patient clustering simulations to compare the methods. For each data set, we applied PAM with Euclidean distance and $k = 2, 3, \ldots, 16$ and evaluated each of the four criteria functions and also MSS. We repeated the simulation independently ten times and kept track of the estimated number of clusters $\hat{K}$ according to each method for each data set.

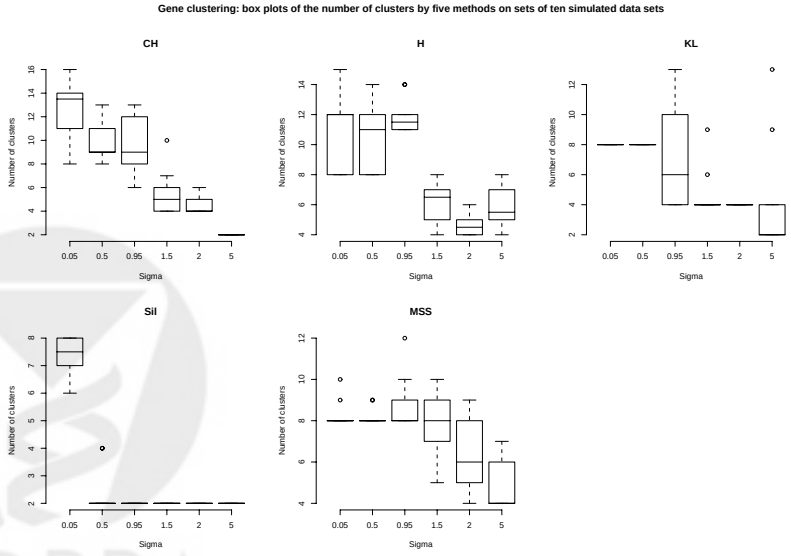

Figure 6: Gene Clustering Simulation - Estimated number of clusters $\hat{K}$ is plotted for a range of values of σ for each criteria.

Figure 6 shows box plots of $\hat{K}$ for each method and each value of σ in the gene clustering simulation. All of the methods show a decrease in $\hat{K}$ with increasing σ. The methods of both Calinski & Harabasz and Hartigan frequently estimate more than the correct number of clusters, while the Silhouette method estimates too few clusters. The method of Krzanowski & Lai performs relatively well, although it makes the switch from eight to four clusters at a lower value of sigma than expected based on the distance between the cluster means. The minimum MSS method performs the best, but shows some variation in $\hat{K}$ over the ten simulations.

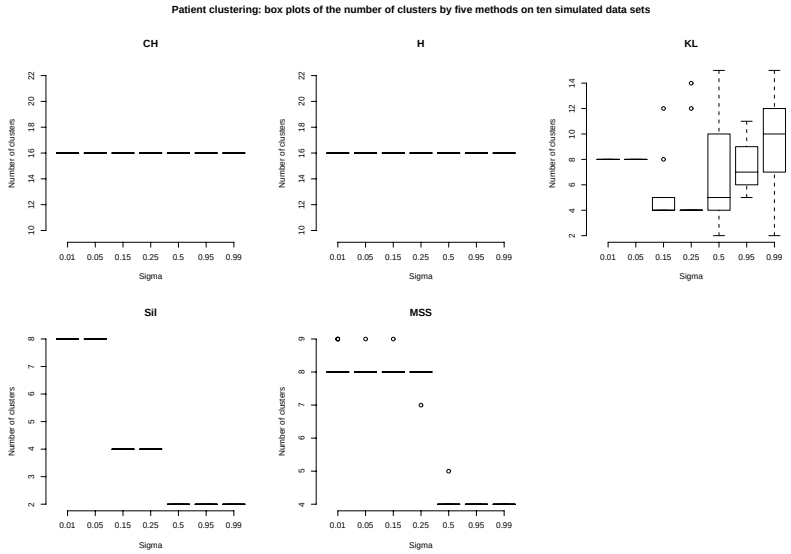

Figure 7: Patient Clustering Simulation - Estimated number of clusters $\hat{K}$ is plotted for a range of values of σ for each criteria.

Figure 7 shows box plots of $\hat{K}$ for each method and each value of σ in the patient clustering simulation. The methods of both Calinski & Harabasz and Hartigan always estimate more than the correct number of clusters, choosing the user-specified maximum number of clusters ($K = 16$) in every simulated data set. The method of Krzanowski & Lai makes the switch from eight to four clusters at a lower value of sigma than expected based on the distance between the cluster means and then begins to estimate an increasing number of clusters at the higher values of σ. The Silhouette method again estimates too few clusters for most values of σ. The minimum MSS method performs the best, showing less variation in $\hat{K}$ over the ten simulations than in the gene clustering simulation.

More extensive simulations, for example using the eight simulated models in [5], are needed to illustrate the relative performance of MSS in a wider range of contexts.

5 Clustering Routine and Criteria

Given a data generating distribution, different clustering routines may produce different clusters. The clusters one would observe if the true distance matrix were known is the clustering parameter, which is estimated by applying the clustering algorithm to the observed distance matrix. In real data analyses, we do not know the clustering parameter, but by studying the performance of different clustering routines on simulated data we can learn about the type of parameters they estimate (e.g.: identifying efficient versus robust methods). Similarly, different criteria for selecting the number of clusters will define different parameters of the true data generating distribution. Hence, it is useful to examine how different criteria for selecting the number of clusters perform on simulated data. In particular, it will be useful to look at how different criteria operate in conjunction with a range of clustering algorithms. There may be some criteria that are better suited to be used with certain algorithms. Ideally, one would like to have a criteria that is optimal independent of the algorithm.

We have presented a general method for selecting the number of clusters which can be implemented with any global criteria function. The method has been illustrated with the criteria silhouette. The method can be applied to any clustering routine, and we have chosen to use PAM and HOPACH as examples of clustering routines. Some preliminary analyses with real data (not reported here) indicate that in situations where it is much harder to cluster the data than in the idealized simulations described in this paper, it might be useful to use the same criteria function in the clustering routine as in the method to choose the number of clusters. For example, PAM (as a partitioning method and as used in HOPACH) minimizes the sum of distances to the closest medoid, rather than maximizing the sum (or average) of silhouettes. Hence, the clusters produced by PAM may not be ideal with respect to minimizing MSS. We have developed a clustering algorithm called PAMSIL that replaces the criteria function in PAM with average silhouette [15]. Since PAMSIL optimizes average silhouette, it may be a more appropriate algorithm to use with MSS. We are currently investigating this idea.

6 Discussion

The minimum MSS method presented in this paper is an implementation of a general approach for selecting the number of clusters. The approach takes a distance matrix, a global criteria (such as average silhouette), and a clustering algorithm as input and returns optimal cluster labels. The idea is to evaluate each of a series of potential clustering results by diving into each of the proposed clusters individually, applying the clustering algorithm to the elements in that cluster, and evaluating the criteria for that cluster. In this way, a measure of heterogeneity is computed for each proposed cluster. By averaging these over clusters, the proposed clustering result can be evaluated

in terms of cluster homogeneity. The clustering result with minimum average heterogeneity is selected. This approach can be used to select the number of clusters with both partitioning and hierarchical algorithms. In its most general form, the method can be used with any clustering routine and any global criteria, although we believe that some thought should be put in to pairing the criteria and the algorithm so that they are optimizing some what similar functions of the data.

We have compared MSS with four of the best performing direct methods in the literature. This comparison was done on simulated data sets with a nested cluster structure similar to patterns we have seen in real gene expression data. We considered both clustering of genes and clustering of samples (e.g.: patients) within clusters of genes. While this simulation is somewhat idealized, it illustrates the challenge of identifying smaller clusters and going beyond global clustering patterns. The minimum MSS method is better able to identify the finer structure in the data than any of the other four methods. In particular, average silhouette tends to only find the two largest clusters even for very low noise levels. In addition, MSS is defined for only one cluster and is indeed minimized at $k = 1$ clusters when the data are unimodal.

An alternative approach to selecting the number of clusters involves testing for evidence against a given null distribution. Testing methods generally involve re-sampling or permuting the data many times and are therefore much more computationally intensive that direct methods. For this reason, we did not compare our method with testing approaches. An extension of this study would be to evaluate a number of methods on a wider selection of models, such as those used in [5].

Although it was developed in the context of gene expression data analysis, the MSS criteria has a much wider range of applications. The nested data structure discussed here occurs in many contexts, and the simulation results we have presented indicate that MSS may be a valuable model selection tool for selecting the number of clusters with both partitioning and hierarchical clustering algorithms.

References

[1] M. Bittner, P. Meltzer, Y. Chen, Y. Jiang, E. Seftor, M. Hendrix, M. Radmacher, R. Simon, Z. Yakhini, A. Ben-Dor, N. Sampas, E. Dougherty, E. Wang, F. Marincola, C. Gooden, J. Lueders, A. Glatfelter, P. Pollock, J. Carpten, E. Gillanders, D. Leja, K. Dietrich, C. Beaudry, M. Berens, D. Alberts, V. Sondak, N. Hayward, and J. Trent. Molecular classification of cutaneous malignant melanoma by gene expression profiling. *Nature*, 406:536–540, 2000.

[2] J. Bryan, K.S. Pollard, and M.J. van der Laan. Paired and unpaired comparison and clustering with gene expression data. *Statistica Sinica*, 2002. To Appear.

[3] M. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci.*, 95:14863–14868, 1998.

[4] J. Fridlyand. Ph.d thesis. Department of Statistics, UC Berkeley, 2001.

[5] J. Fridlyand and S. Dudoit. Applications of resampling methods to estimate the number of clusters and to improve the accuracy of a clustering method. Technical Report 600, Statistics Department, University of California, 2001.

[6] L. Kaufman and P.J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, New York, 1990.

[7] G.W. Milligan and M.C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50:159–179, 1985.

[8] K.S. Pollard and M.J. van der Laan. Statistical inference for simultaneous clustering of gene expression data. *Mathematical Biosciences*, 176(1):99–121, 2002.

[9] K. Rose, E. Gurewitz, and G.C. Fox. Statistical mechanics and phase transitions in clustering. *Physical Review Letters*, 65:945–948, 1990.

[10] A.J. Scott and M.J. Simmons. Clustering methods based on likelihood ratio criteria. *Biometrics*, 27:387–397, 1971.

[11] R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a dataset via the gap statistic. Technical report, Department of Statistics, Stanford University, March 2000.

[12] P. Törönen, M. Kolehainen, G. Wong, and E. Castren. Analysis of gene expression data using self-organizing maps. *FEBS Letters*, 451:142–146, 1999.

[13] M.J. van der Laan and J.F. Bryan. Gene expression analysis with the parametric bootstrap. *Biostatistics*, 2:1–17, 2001.

[14] M.J. van der Laan and K.S. Pollard. Hybrid clustering of gene expression data with visualization and the bootstrap. Technical Report 93, Group in Biostatistics, University of California, May 2001. To appear in JSPI.

[15] M.J. van der Laan, K.S. Pollard, and J. Bryan. A new partitioning around medoids algorithm. Technical Report 105, Group in Biostatistics, University of California, February 2002. Submitted.