Statistica Sinica **12**(2002), 87-110

PAIRED AND UNPAIRED COMPARISON AND CLUSTERING WITH GENE EXPRESSION DATA

Jenny Bryan[1], Katherine S. Pollard[2] and Mark J. van der Laan[2]

[1] *University of British Columbia and* [2] *University of California, Berkeley*

Abstract: We have previously described a statistical framework for using gene expression data from cDNA microarrays to select meaningful subsets of genes and to place genes into clusters (van der Laan and Bryan (2001)). In this paper we extend the methodology to the setting in which expression data is collected on a common set of p genes from either two observations within a subject (paired), or on subjects from two subpopulations (unpaired). We present simulation results that illustrate important issues encountered with cluster analysis in gene expression data. In particular, we see that sampling variability of the covariance structure and the presence of unrelated genes can have a strong impact on clustering algorithms and measures of cluster strength. We discuss ways to address this issue, including the application of a hybrid clustering method which incorporates both partitioning and collapsing steps. The hybrid methodology is illustrated on a cancer cell line data set with two types of cancer. We also present a method for selecting significantly differently expressed genes using a null distribution. Finally, we present theoretical results relating to sample size and consistency in this setting.

Key words and phrases: Bootstrap, cluster analysis, gene expression analysis.

1. Introduction

1.1. Context

Microarrays allow researchers to monitor the intensity of gene expression for thousands of genes simultaneously. Since these experiments have been described well elsewhere, we refer the interested reader to other sources of information on the experiment itself. Good overviews can be obtained from http://www.cs.washington.edu/homes/jbuhler/research/array/ and Marshall (1999); a more technical description is provided by the articles in the "The Chipping Forecast" (1999).

The growing use of microarrays in biological research has created the need for new statistical methods that are tailored to the specific research questions addressed and that accommodate typical features of the data structure. Major areas of emphasis include clustering and classification, both of genes and of subjects. The application of clustering techniques to gene expression data was

first described in Eisen, Spellman, Brown and Botstein (1998). An alternative to cluster analysis, called "gene-shaving", is proposed by Hastie et al. (2000). Golub et al. (1999) propose a method for classifying leukemia patients based on microarray data. Various techniques for classifying experimental units are compared in Dudoit, Fridlyand and Speed (2000). The methods described in this paper are aimed at finding subsets and clusters of genes when the subjects are either drawn from the same population and observed at two time points (paired comparison), or drawn from two subpopulations (unpaired comparison).

1.2. Statistical framework

By definition, array-based technologies, such as cDNA microarrays, provide gene expression data on a very large number of genes at once. An expression vector $\mathbf{X} = (X_1, \ldots, X_j, \ldots X_p)$, consisting of p ratios, is the fundamental unit of data. In this subsection, we first consider the one sample problem where we observe n i.i.d. copies of $\mathbf{X}$. Typically, the expression vector is transformed to the log scale and truncated. Investigators often seek a subset of genes, much smaller than the full set of p genes, that exhibit certain meaningful patterns of expression. We call this subset the *target subset* $\mathcal{S}$ and the mapping which produces it a *subset rule*. A typical subset rule will draw on "screens" and "labellers". A screen is used to eliminate certain genes from the subset. For example, one might retain only differently expressed genes. In Section 3 we discuss methods for designing such a screen using a null distribution. A labeller will apply labels, such as the output of a clustering routine. Clustering routines commonly applied to gene expression data include partitioning algorithms (k-means, self-organizing maps) and agglomerative algorithms (hierarchical single or average linkage methods). Meaningful analyses can be done with various combinations of screens and labellers or even with a screen or labeller alone. The target subset $\mathcal{S}$ can be encoded in a p-vector in which $\mathcal{S}_j = 0$ implies that gene j is not in the target subset and $\mathcal{S}_j = k$ implies that gene j is in the target subset and belongs to cluster k, where $j \in \{1, \ldots, p\}$ and $k \in \{1, \ldots, K\}$. The target subset $\mathcal{S}$ (with cluster labels) is the subset we would select if the true data generating distribution were known, and it is estimated by the observed sample subset $\widehat{\mathbf{S}}_n$.

We now define a number of quality measures for the estimated subset $\widehat{\mathbf{S}}_n$, which measure the deviation of $\widehat{\mathbf{S}}_n$ from the target subset $\mathcal{S}$. Imagine a subset of genes that lie far from the target subset $\mathcal{S}$ and that should appear in the estimated subset $\widehat{\mathbf{S}}_n$ rarely, if ever. When such a gene appears in $\widehat{\mathbf{S}}_n$, we call it an "extremely false positive". We define PEFP as the proportion of genes in the estimated subset $\widehat{\mathbf{S}}_n$ which are extremely false positives and PAFP as the probability that the subset $\widehat{\mathbf{S}}_n$ contains any extremely false positives. If we think

of $\widehat{\mathbf{S}}_n$ as a "screening test" for $\mathcal{S}$, it is natural to adapt the notions of sensitivity and positive predictive value as measures of the overall quality of the estimated subset $\widehat{\mathbf{S}}$. Specifically, $sens = sens_n = |\mathcal{S} \cap \widehat{\mathbf{S}}|/|\mathcal{S}|$, $ppv = ppv_n = |\mathcal{S} \cap \widehat{\mathbf{S}}|/|\widehat{\mathbf{S}}|$, where $|\cdot|$ denotes cardinality of a set. We define cluster-specific sensitivity and positive predictive value in a similar manner. The expected values of these proportions represent quality measures of interest.

For a fixed subset rule, sample size n, and true data-generating distribution, each gene j has some fixed probability $p_{j,n} \equiv P(\widehat{\mathbf{S}}_j > 0)$ of appearing in the estimated subset $\widehat{\mathbf{S}}_n$ and if the subset rule applied cluster labels, some fixed probability $p^k_{j,n} \equiv P(\widehat{\mathbf{S}}_j = k)$ of appearing in the estimated subset carrying label k. As the sample size grows, these probabilities approach 1 if gene j is in $\mathcal{S}$ (with appropriate label for $p^k_{j,n}$) and 0 otherwise. Knowledge of these reappearance probabilities provides a basis for ranking the genes based on the strength of evidence that gene j is in $\mathcal{S}$ or carries a certain label.

To estimate the cluster quality measures and reappearance probabilities described above, we require knowledge of the sampling distribution of $\widehat{\mathbf{S}}_n$. van der Laan and Bryan (2001) employ a parametric bootstrap, using a multivariate normal model and discuss the implications of this choice of model. Pollard and van der Laan (2001) conduct simulations which illustrate that for smooth mappings such as continuous functions of sample means, the computationally easier nonparametric bootstrap performs as well as the parametric bootstrap under a normal model. Hence, use of the non-parametric bootstrap is a good way to protect against misspecification of the parametric model.

1.3. Application to paired and unpaired comparisons

Now suppose we wish to compare two sets of relative gene expression measurements $(\mathbf{X}, \mathbf{Y})$ on a common set of p genes. Such data can arise under two different scenarios: paired and unpaired. In the paired scenario, we have two observations on each subject. For example, gene expression might be measured on a cell line at two different time points in the cell cycle relative to a baseline. Or, we might observe the same subject before and after treatment. Perou et al. (2000), for example, analyzed gene expression in human breast cancer tumors before and after chemotherapy using a common reference sample. In the unpaired scenario, we have observations on subjects drawn from two subpopulations (possibly with different numbers of observations in each subsample). Golub et al. (1999), for example, used gene expression data to distinguish between acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML).

We might want to focus on genes that appear to be very differently expressed in the two data sets. One approach is to simply analyze the two data sets separately and compare the clustering patterns. Another approach is to combine

the two data sets into one data set. The way we do this depends on how the data were generated. In the paired scenario, we can form a p-dimensional vector of log ratios, $\log(\mathbf{X}_i/\mathbf{Y}_i)$, by dividing the relative expression for a subject at one time point by that at the other before taking the log. In the unpaired scenario, we can form a p-dimensional vector of log ratios by dividing the relative expression for a subject by the geometric mean relative expression for all subjects in the other subpopulation before taking the log so that we get $\log(\mathbf{X}_i/\hat{\boldsymbol{\mu}}_Y)$. In both cases, the empirical mean of the combined data set is the difference between the two sample means of the log ratios in the separate data sets.

1.4. Organization of the paper

First, we present simulations designed to establish the behavior of $\widehat{\mathbf{S}}_n$ for several subset rules $\mathcal{S}$ involving pre-screening, clustering, and post-screening of genes. In addition, we use these simulations to establish the practical performance of the bootstrap in estimating the variability of $\widehat{\mathbf{S}}_n$. The clustering is implemented with the algorithm called "partitioning around medoids" (PAM) (Kaufman and Rousseeuw (1990), Chap. 2). Second, we present a data analysis which uses new methods for (i) selecting differently expressed genes, and (ii) identifying groups of correlated genes through clustering. These new methods address issues identified in the simulation. Finally, we present consistency and sample size results.

2. Simulation

The goal of this study is to explore the performance of $\widehat{\mathbf{S}}_n$ and the bootstrap in the context of a known data-generating distribution. We are particularly interested in assessing the difficulty of applying cluster labels in the presence of genes that belong to no cluster, and of how that is affected by sample size. The simulation shows that it is beneficial to screen unrelated genes prior to applying a clustering algorithm. We also see that unrelated genes tend to depress conventional measures of the clustering strength. Lastly, it is apparent that post-screens affected by isolated extreme values, such as the smallest entries in a column of a correlation matrix, will require large sample sizes to achieve good performance of $\widehat{\mathbf{S}}$, and alternative screens should be considered.

2.1. Data-generating distribution

We create a data-generating distribution by assuming a multivariate normal model and selecting the parameters $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. We take $K = 3$ clusters – cluster A, cluster B, and cluster C – each containing 100 genes. Each cluster has a core set of genes that are more highly correlated with one another, and a more

weakly correlated set of peripheral genes. The genes in a given cluster have no correlation to genes in the other clusters. The clustered genes are embedded in a set of 300 other genes that have absolutely no correlation with other genes at all. The correlation matrix of the full set of 600 genes, $\boldsymbol{\rho}$, is block diagonal. With this set-up we are trying to simulate what seems to be an important data structure: a fraction of the genes being studied are involved in the phenomenon of interest and even break down into several well-defined clusters, but there are many "noisy" genes on the array, which are not involved and whose presence makes it difficult to find the relevant clusters.

The mean expression levels are also set with the cluster structure in mind. The noisy genes have means near zero, with some individual genes exhibiting a mild amount of differential expression. Cluster A contains genes that are over-expressed, many quite strongly. Most genes in Cluster B are differently expressed, with slightly more being under-expressed than over-expressed. Cluster C contains genes with a wide range of expressions. Gene-specific standard deviations have different distributions for each cluster and for the noisy genes. Throughout this section, we use yellow for cluster A, violet for cluster B, and blue for cluster C.

2.2. Subset rule

The subset rule is applied to the true mean and covariance (not simulated data), so that we can examine properties of the target subset $\mathcal{S}$. The rule we use is typical of those applied in microarray data analyses: first, screen for differently expressed genes and then apply cluster analysis. We exclude genes with an absolute mean $|\mu_j| < \log_2 1.5 = 0.58$, which corresponds to 1.5-fold differential expression. Of the 600 genes, 318 are retained and 282 are excluded based on this screen.

The remaining 318 genes are provided to a cluster analysis routine, with the dissimilarity between two genes defined as 1 minus the absolute value of the correlation. For a fixed number of clusters K, a partitioning method finds the best grouping and, by exploring different values of K, we can assess the evidence for different K values (see Kaufman and Rousseeuw (1990), Chap. 1, Sect. 3). It is also valuable to assign meaning to a particular cluster label k as most scientific papers that employ cluster analysis to analyze microarray data discuss the unifying theme of the genes found in each cluster. In the context of one data set, any clustering algorithm will likely yield at least one partition that can be interpreted. However, when one views a data set and its clustering as just one realization of a stochastic phenomenon, it is desirable to have a way to enforce a coherent meaning for cluster label k. The cluster centers that are important in most partitioning methods play this role very well. By fixing cluster centers, one can ensure it is sensible to compare genes with label k from one realization

of the experiment to the next. Lastly, we prefer "partitioning around medoids" (PAM) to k-means because we like being able to use any distance metric, and we prefer that cluster centers be one of the underlying objects (in this case, a gene) instead of an average of objects, a quantity that is difficult to interpret and less robust to outliers.

Given any partition, Kaufman and Rousseeuw (1990) define for each object a quantity called the silhouette, which reflects how well-matched an object is in its cluster versus the next closest cluster. Silhouettes take values in the interval $[-1, 1]$, with 1 corresponding to a perfect match. Silhouettes are a valuable tool for assessing what is basically the goodness-of-fit for a clustering. For a given data set and clustering method, silhouettes can be compared for different numbers of clusters in order to choose the optimal number. There were 251 genes in clusters A, B, and C that passed the differential expression screen ($318 - 251 = 67$ noise genes pass the screen, but have silhouettes of zero). Silhouettes were examined for $K = 2, 3$, and 4. When $K = 2$, we see that cluster B is fully recovered, while clusters A and C are lumped together. When $K = 3$, which we know to be the correct value of K, we see that PAM recovers the underlying clusters. When $K = 4$, clusters A and C are fully recovered and Cluster B is split into two. The lack of evidence for $K = 4$ is apparent in the erratic, even negative, silhouettes for genes in cluster B. The core versus periphery structure of the underlying clusters is also reflected in the silhouettes. Table 1 presents average silhouettes for these clusterings, with and without the 67 noise genes that pass the differential expression screen. We see that the overall average silhouette is highest at the correct value of K, which is 3, regardless of the presence of the noise genes. But the noise genes have a dampening effect on the silhouettes in general. This points out the benefit of eliminating all unrelated genes prior to attempting any type of cluster analysis.

Table 1. Average Silhouettes for $K = 2, 3, 4$ in simulation study.

Which genes?	K	Overall Avg. Silh.	Cluster 1	Cluster 2	Cluster 3	Cluster 4
318	2	0.09	0.06	0.17		
genes	3	0.13	0.12	0.17	0.11	
(67 noise)	4	0.09	0.12	0.02	0.04	0.11
251	2	0.07	0.03	0.17		
genes	3	0.09	0.04	0.17	0.10	
(no noise)	4	0.05	0.04	0.06	0.02	0.10

After clustering the genes, we chose to apply one last screen in another attempt to eliminate uninteresting genes. The goal is to remove genes that are not

particularly well-matched to their cluster. We used two different approaches, one based on pairwise dissimilarities and one based on silhouettes. The dissimilarity screen DYS works as follows: the cluster center (or "medoid") is automatically included; any gene with a dissimilarity of less than 0.655 with the cluster center is included; any gene with a dissimilarity of less than 0.655 with any previously included gene is also included; this last step is repeated until no changes occur. The silhouette screen SILH includes all genes with a silhouette greater than 0.08. Both screens result in target subsets $\mathcal{S}$ containing 150 genes. Table 2 presents target subset $\mathcal{S}$ membership by true cluster membership for both screens.

Table 2. Target subset membership by true cluster membership.

	Target Subset $\mathcal{S}_j$ =					
True	0	1	2	3	> 0	All
DYS screen						
Noise	300				0	300
Cluster A	50	50			50	100
Cluster B	27		73		73	100
Cluster C	73			27	27	100
	450	50	73	27	150	600
SILH screen						
Noise	300				0	300
Cluster A	71	29			29	100
Cluster B	12		88		88	100
Cluster C	67			33	33	100
	450	29	88	33	150	600

To summarize the subset rule, the genes were first screened for differential expression by requiring that $|\mu_j| > 0.58$. The remaining 318 genes are clustered by PAM, with the cluster number $K = 3$. The cluster centers are noted and remain fixed in future analyses. In light of the clustering, genes are screened again based either on dissimilarities or silhouettes to yield a final subset containing 150 genes and their cluster labels.

2.3. Sampling distribution of $\widehat{\mathbf{S}}_n$

We generated 100 samples of size $n = 25, 50$, and 150 from the chosen data-generating distribution $N((\boldsymbol{\mu}, \boldsymbol{\Sigma}))$ and applied the two subset rules described above. Based on these samples, we can estimate the reappearance probabilities p_j and p_j^k. In Figure 1, we examine the effect of sample size in the DYS screen. The results are somewhat counter-intuitive but illustrate an important phenomenon. At the series of sample sizes considered here, overall p_j tend to decrease for all

genes. Average p_j within different values of $\mathcal{S}$ are presented in the lower left panel. It is important to examine the cluster-specific reappearance probabilities. The top panel presents this information for 4 typical genes, one for each value of $\mathcal{S}$, and the lower right panel presents averages within values of $\mathcal{S}$. We see that, while overall p_j may be declining, the correct cluster-specific p_j^k are climbing steadily. One expects that, had we added a larger sample size such as $n = 300$, even the overall p_j would begin to increase as n does.

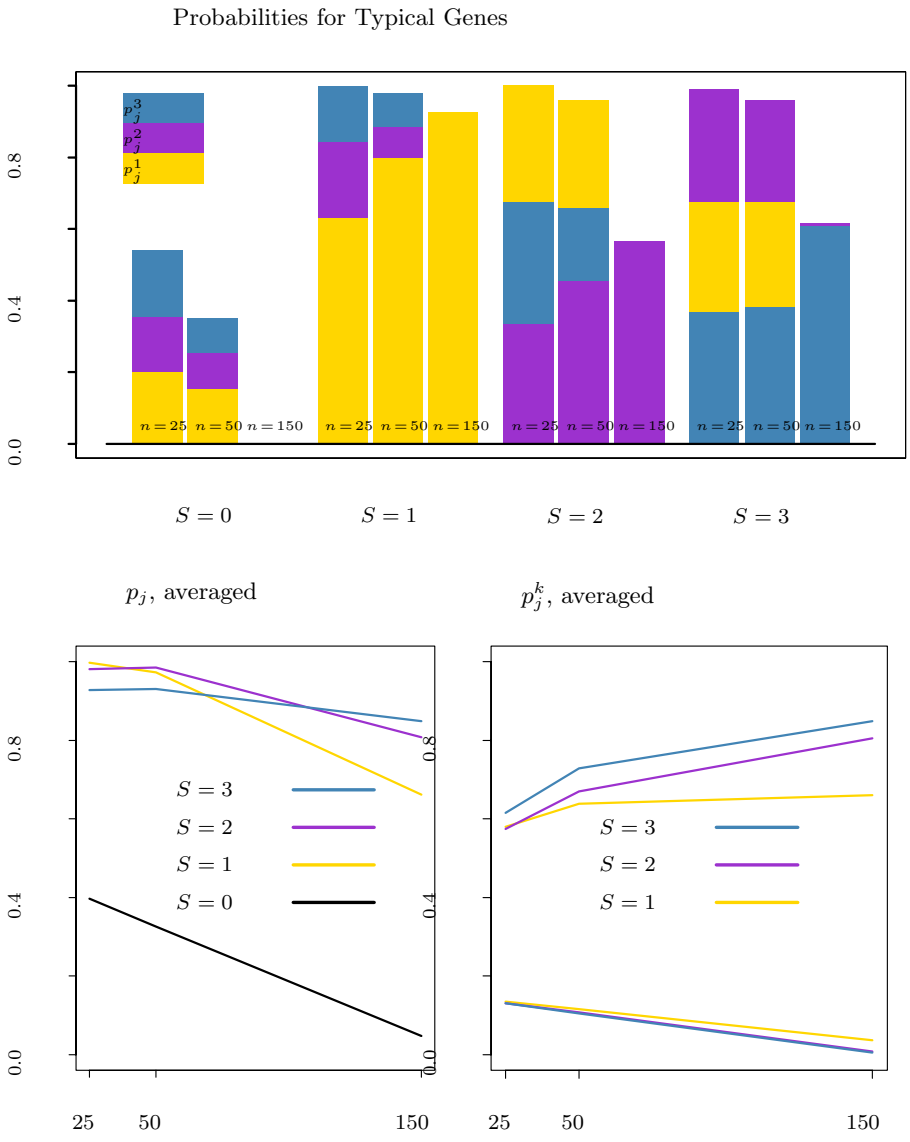

Figure 1. Reappearance probabilities, DYS screen in simulation study.

The results of this simulation demonstrate that the mean requires much less data to estimate than the covariance structure. For all sample sizes, the expected number of genes passing the differential expression screen is very close to the true number of 318. It is approximately 325, 323, and 321 for $n = 25, 50$, and 150, respectively. From other simulations not reported here, in which the subset rule consists solely of the differential expression screen, we know that both sensitivity and positive predictive value at this stage are extremely high (between 0.95 and 0.99) and, therefore, the *correct* genes are almost always passing this initial screen at all sample sizes. The problem occurs in the clustering and DYS screen – that is, the steps of the subset rule that depend on the covariance. At $n = 25$, many genes are misclassified but frequently pass the dissimilarity screen due to sampling variability in the covariance. Since a gene can pass this screen by exhibiting even one extremely small pairwise distance, it is almost always passed for small samples. Therefore, the probability of appearing in the $\widehat{\mathbf{S}}_n$ has significant contributions from all three cluster-specific probabilities p_j^1, p_j^2, and p_j^3. This can be seen in the first stacked column for each of the 4 genes highlighted in the top panel of Figure 1. As the sample size increases to 50 and 150, misclassification decreases and p_j becomes dominated by the correct cluster-specific probability. This can be seen in the second and third stacked columns. These simulation results suggest a modification of the DYS screen in which a gene must have a sufficiently small dissimilarity with the cluster center.

The behavior described above is also apparent in subset-wide measures of quality, reported in Table 3. As expected, sensitivity decreases at these sample sizes, but the positive predictive value increases. Once again, we conjecture that the sensitivity would increase for $n > 150$. Extremely false positives were defined as genes with absolute mean expression less than $\log_2 1.1 \approx 0.14$. The expected proportion of extremely false positives (E{PEFP}) is essentially zero for all n and the probability of any extremely false positives (PAFP) decreases as n grows.

Table 3. Cluster-wide quality measures for the DYS rule in the simulation study.

	$n = 25$	$n = 50$	$n = 150$
E{Sens}	0.98	0.97	0.77
E{PPV}	0.45	0.50	0.84
E{PEFP}	0.00	0.00	0.00
PAFP	0.48	0.09	0.00

The situation is quite different for the subset rule SILH that screens based on the silhouettes. Summary information on p_j and p_j^k is depicted graphically in Figure 2. It is immediately apparent that the reappearance probabilities are much lower in general than those seen with the DYS rule. This is due to the fact

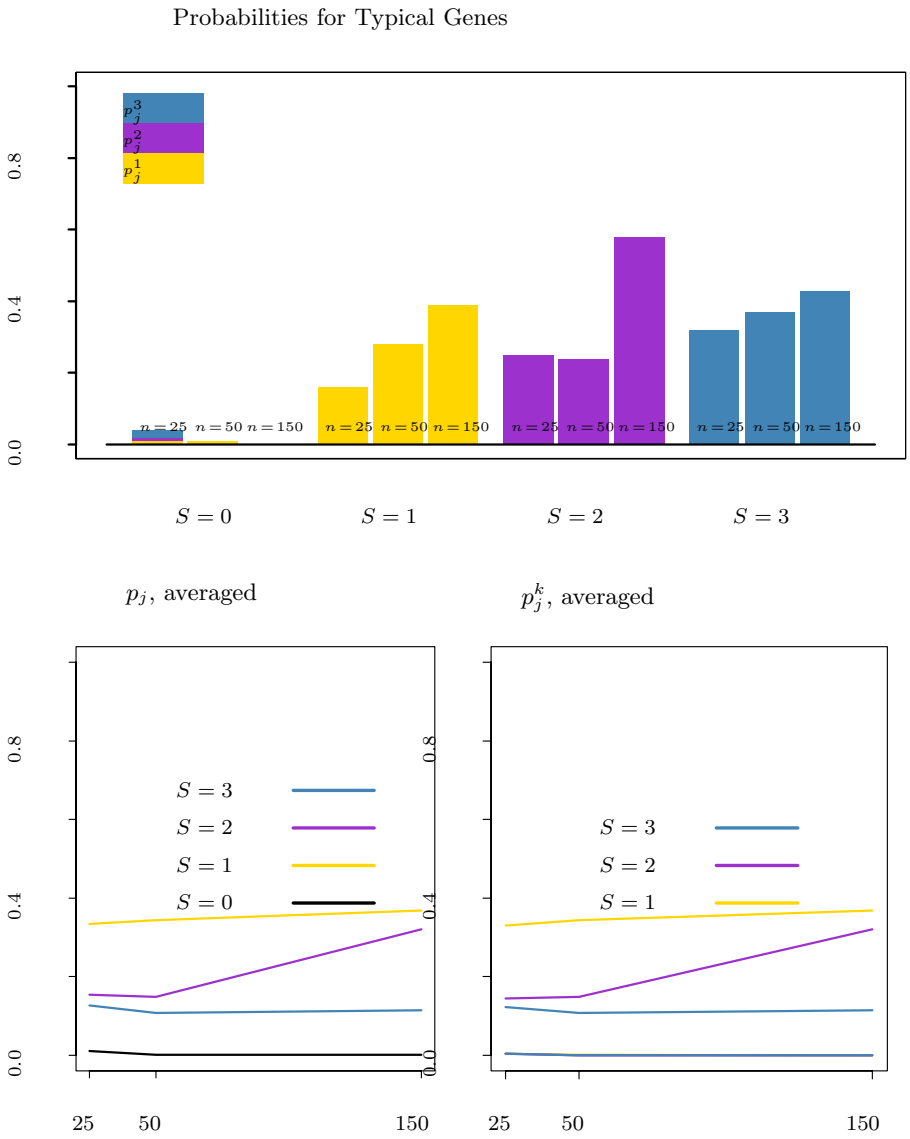

Figure 2. Reappearance probabilities, SILH screen in simulation study.

that, compared to the silhouettes produced by the true block diagonal correlation matrix, silhouettes in observed data are lower. The average silhouette in the target subset $\mathcal{S}$ is 0.09. The expected average silhouette in the sample subset $\widehat{\mathbf{S}}$ is 0.02. The non-zero empirical correlation that arises between even unrelated genes has the effect of making the clustering appear to be less strong. Therefore, when applying the silhouette cutoff to a clustering based on a finite amount of data, we are left with a smaller set of genes. The average size of $\widehat{\mathbf{S}}_n$ is approximately

32, 27, and 43 for $n = 25, 50$, and 150, respectively. Both the expected subset size and the p_j and p_j^k seem to grow very slowly as the sample size increases. We have also noted here and in other analyses that the values of the silhouettes are very dependent on the dimension of the data set (number of genes), so that universal cutoff values as described in Kaufman and Rousseeuw (1990) are not appropriate in the gene expression context. One screen that may be more useful than absolute cutoffs based on silhouettes is to always retain a fixed number of top-ranked genes based on silhouettes or estimated cluster-specific probabilities. If one wishes to test the significance of a silhouette, we propose using a simulation from an appropriate null distribution (i.e., one with no clustering).

Table 4 presents subset-wide measures of quality for the SILH rule. Both sensitivity and positive predictive value increase with n and the probability of any false positive is extremely small even at $n = 25$ and quickly falls to zero.

Table 4. Cluster-wide quality measures for the SILH rule in the simulation study.

	n =25	n =50	n =150
E{Sens}	0.18	0.18	0.28
E{PPV}	0.86	0.98	0.99
E{PEFP}	0.00	0.00	0.00
PAFP	0.04	0.00	0.00

We are also interested in the actual gene-specific probabilities p_j and p_j^k. For genes in $\mathcal{S}$ for the DYS rule, although the overall p_j decrease, the correct cluster-specific probabilities p_j^k increase with n. In fact, at $n = 150$, essentially no genes appear in $\widehat{\mathbf{S}}$ carrying the incorrect cluster label. This observation supports the above discussion of the DYS rule. Consistent with the above findings regarding the stringency of the silhouette-based screen, we see relatively low p_j, which grow very slowly with n, for the SILH rule. The misclassification of genes is practically impossible with this rule.

2.4. Bootstrap results

For each of the 100 size samples generated from the data-generating distribution $N((\boldsymbol{\mu}, \boldsymbol{\Sigma}))$, we carried out the parametric bootstrap as described in van der Laan and Bryan (2001). Since the simulated data is multivariate normal distributed here, the use of this distribution in the bootstrap is appropriate. The empirical distribution of the bootstrap subsets allows us to estimate interesting features of the sampling distribution of $\widehat{\mathbf{S}}$. The probability of gene j appearing in $\widehat{\mathbf{S}}_n$, i.e., p_j, is estimated by the proportion of bootstrap subsets in which gene j appears. An analogous approach leads to estimates of p_j^k. Figures 3 and 4 plot

true reappearance probabilities against average bootstrap probabilities for the DYS and SILH rules, respectively.

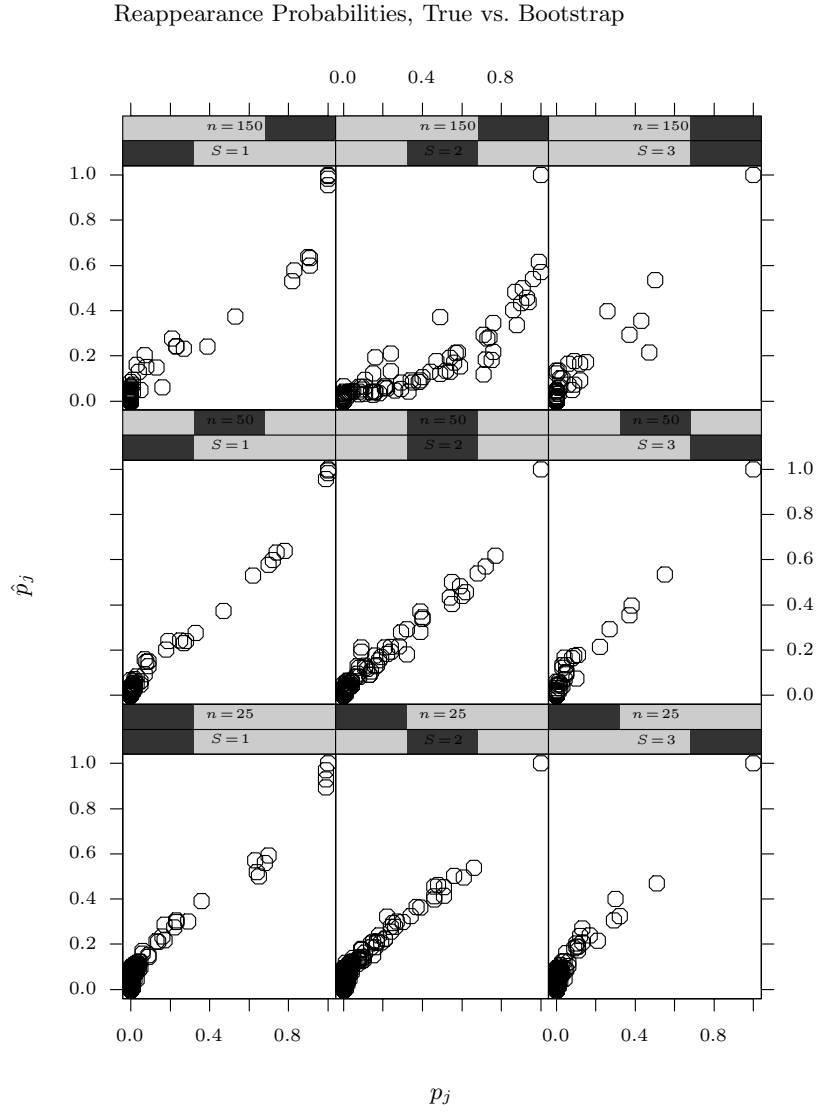

Figure 3. True reappearance probabilities vs. averages from bootstrap, DYS screen in simulation study.

In finite samples, the expected bootstrap probabilities are biased estimators of the true probabilities. For certain simple rules, this bias is relatively straightforward to quantify and is discussed in van der Laan and Bryan (2000). For

complicated rules such as DYS and SILH, the only relevant result is that, as $n \to \infty$, the expected bootstrap probabilities will approach 1 for genes in $\mathcal{S}$ and 0 for all other genes. Graphically, this means that, as $n \to \infty$, we will eventually see points only at (0,0) and (1,1). But for finite n, plots such as those in Figures 3 and 4 are the best way to understand the relationship between the expected bootstrap and true reappearance probabilities.

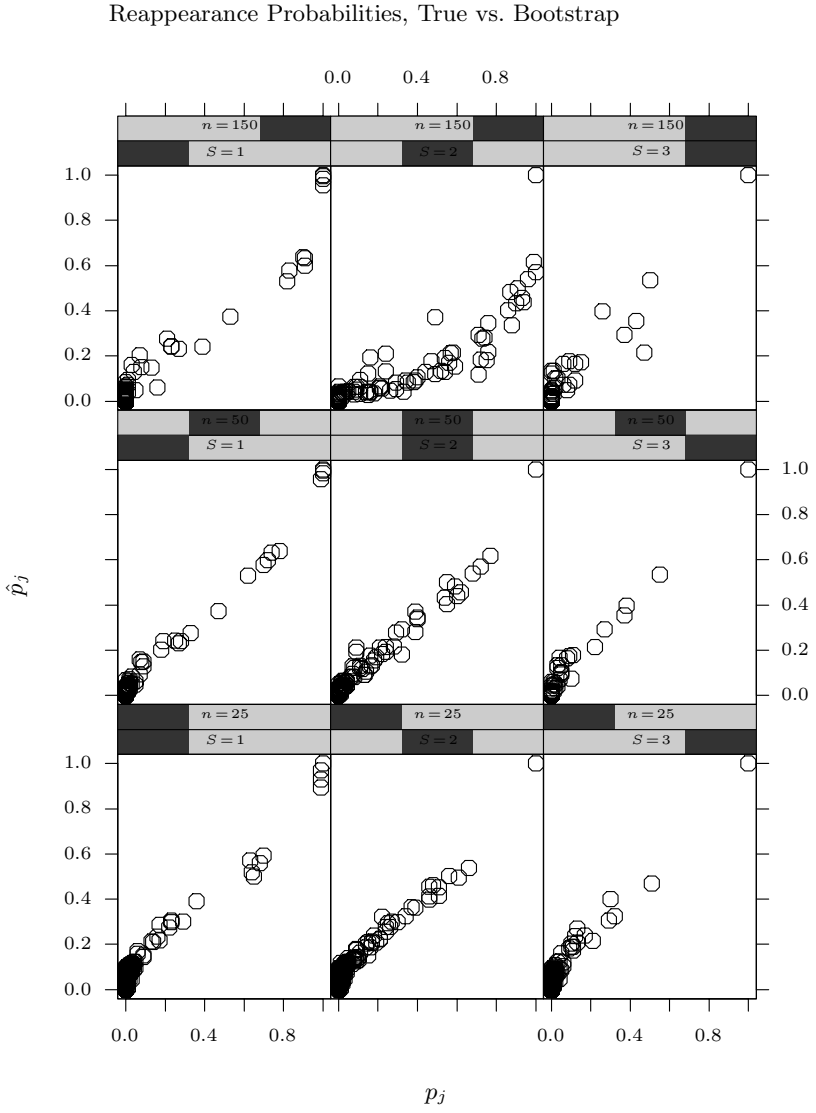

Figure 4. True reappearance probabilities vs. averages from bootstrap, SILH screen in simulation study.

2.5. Distribution of the sample mean

For a fixed n and δ, the formula given as equation 1 in Section 4 can be solved for the ϵ such that the probability of even one component of the p-dimensional sample mean $\hat{\boldsymbol{\mu}}_n$ varying by more than ϵ from the corresponding component of the true mean $\boldsymbol{\mu}$ is less than δ, $0 < \delta < 1$. The sample size is quite conservative, since it does not exploit the correlation among the genes. That is, when one computes values of $\epsilon > 0$ as described below, the actual probability of $\max_j |\hat{\mu}_j - \mu_j| > \epsilon$ is much less then δ. Table 5 illustrates this and also shows that one can use a value of σ that is much smaller than the actual maximum of the gene-specific log ratio standard deviations and still see favorable results. In all instances, $n = 25$ and $M = 5$.

Table 5. Demonstration that the sample size formula is conservative.

Nominal δ	ϵ	Actual δ	σ
0.05	2.64	0.000	$\max_j \sigma_j = 2.06$
0.50	2.27	0.000	$\max_j \sigma_j = 2.06$
0.20	1.52	0.005	75-th quantile of $\sigma_j = 0.89$
0.40	1.14	0.030	25-th quantile of $\sigma_j = 0.37$
0.80	1.01	0.055	0.25

An alternative, less conservative approach to determining the sample size needed for a certain precision is to perform simulations utilizing the correlation structure in the data. By the Central Limit Theorem, we have that the sample mean $\hat{\boldsymbol{\mu}}$ is asymptotically distributed $N(\hat{\boldsymbol{\mu}}_n, \hat{\boldsymbol{\Sigma}}_n)$. By simulating from this distribution, we can determine the sample sizes needed for different levels of precision. Non-parametric simulations could also be employed.

2.6. Conclusions

These simulations illustrate some important issues encountered in cluster analysis of gene expression data. In particular, we see that sampling variability of the covariance structure and the presence of unrelated genes can have a strong impact on partitioning algorithms and measures of cluster strength and stability. We have found that pre- and post-screening of the genes helps to avoid some of these problems. The simulations show that screens based on differential expression are accurate even for small sample sizes, whereas screens based on the covariance are harder to estimate accurately. One drawback of screening the genes, however, is that important or interesting genes can be excluded along with the "noisy" genes we wish to remove.

In response to this issue, we have developed an algorithm called Hierarchical Ordered Partitioning And Collapsing Hybrid (HOPACH), which incorporates both partitioning and agglomerative steps in order to identify clustering patterns

in the data even in the presence of many unrelated genes. We have conducted simulations which illustrate that this methodology does better than simple partitioning or agglomerative methods at identifying small clusters in the presence of many noisy genes (van der Laan and Pollard (2001)). In Section 3 we outline the HOPACH method and apply it to a cell line data set with two subpopulations. We also demonstrate methods for selecting differently expressed genes using a null distribution.

3. Data Analysis

We examine a data set which is an example of an unpaired comparison with observations from two subpopulations. We extracted a publicly available data set from the data base accompanying Ross et al. (2000). The authors performed microarray experiments on 60 human cancer cell lines (the NCI60) derived from tumors from a variety of tissues and organs by researchers from the National Cancer Institute's Developmental Therapeutics Program. The data set includes gene expression measurements for 9,703 cDNAs representing approximately 8,000 unique transcripts. Each tumor sample was cohybridized with a reference sample consisting of an equal mixture of twelve of the cell lines chosen to maximize diversity. We used the normalized tumor: reference ratios, as in Ross et al. (2000). These were transformed to a log10 scale and truncated above and below, so that any ratio representing greater than 20-fold over- or under-expression was set to log10(20).

For this comparative analysis, we selected two very different types of cancer from those included in the NCI60: melanoma and breast. We created a data set with all samples from these two types of cancer, which included seven breast and eight melanoma cell lines. Next, we applied an initial subset rule in order to reduce the size of the data set for computational reasons only. We retained those genes where at least 30% of all cell lines had a ratio corresponding with greater than 2-fold over- or under-expression. Using a 30% cut-off, a gene differentially expressed in one type of cancer and not the other would still be included. There were 3500 genes in the resulting data set. This data set was divided into two smaller data sets consisting of the cell lines from each type of cancer. These data sets were analyzed separately and also combined into one data set by dividing the melanoma ratios by the geometric mean breast ratios before taking the log. Unless otherwise noted, we are working with the single, combined data set containing 3500 genes and eight observations.

One goal of the analysis is to identify genes differently expressed in melanoma relative to breast cancer; such genes help us to understand the biological characterization of different cancers and may lead to new cancer-specific treatments.

Another goal is to study clustering patterns in the data set in order to discover information about how the genes involved in tumors work together.

3.1. Selecting differently expressed genes

A common approach to selecting differently expressed genes is to retain those genes whose absolute mean log ratios are greater than some cut-off value. In order to account for variance as well as mean expression, one can standardize the log ratios by dividing them by their gene-specific standard errors before taking the mean. These standardized means can be compared to the quantiles of a standard normal distribution on an individual basis. For the combined data set, $p^* = 1731$ genes were significantly differently expressed at the $\alpha = 0.05$ level (cut-off value= 0.69). Since we are in a multiple comparisons setting, it is advisable to adjust the cut-off value. The Bonferoni adjusted cut-off value was 1.53 and produced a much smaller subset of $p^* = 605$ differently expressed genes.

An alternative, more exact approach is to derive a cut-off value from an appropriate null distribution with zero means and the true covariance structure. A parametric method is to generate a large number of samples from a multivariate normal distribution $N(0, \rho)$, where ρ is the correlation matrix, and select a cut-off value such that no more than $1 - \frac{\alpha}{2}$ of samples have any differently expressed genes. The correlation matrix ρ can be estimated by the empirical correlation matrix. A non-parametric method is to standardize the observed data so that each gene has mean zero and variance one, then generate a large number of bootstrap samples from this data (resampling cell lines with replacement), and use these to compute the cut-off value such that no more than $1 - \frac{\alpha}{2}$ of samples have any differently expressed genes. For both the parametric and non-parametric methods, a less stringent approach is to choose the cut-off value such that on average any sample is expected to have no more than $1 - \frac{\alpha}{2}$ of genes differently expressed. We used the nonparametric bootstrap with the more stringent criteria and obtained a subset of $p^* = 889$ genes. The cut-off value was 1.17, which lies between the value which ignores the multiple comparisons and the too strict Bonferoni adjusted value.

3.2. Clustering genes

HOPACH

We now turn to a discussion of our clustering method. The HOPACH method produces a hierarchical tree of clusters by applying a partitioning algorithm iteratively. A collapsing step which unites the two closest clusters into one cluster can be used at any level of the tree to correct for errors in the number of clusters. For example, we might collapse whenever doing so improves the overall average silhouette for that level of the tree. At each node, a cluster is split into two or

more smaller clusters with an enforced ordering of the clusters and of the elements within clusters. The final level of the tree is an ordered list of the elements. The ordering of elements at any level of the tree can be used to visualize the clustering structure in a colored plot of the reordered data or distance matrix. Visual comparison of the distance matrix for different levels of the tree with the final distance matrix typically identifies the main clustering structure and provides information about the clusters, such as their strength and their similarity to each other. After identifying the clusters, the bootstrap can be used to establish the reliability of these clusters and the overall variability of the followed procedure.

The HOPACH methodology is a general approach which could be applied with any choice of partitioning algorithm. We refer to our particular implementation using PAM for the partitioning algorithm as HOPACH-PAM. HOPACH-PAM can be run with any user-supplied distance metric (euclidean, correlation, absolute correlation). The cosine-angle distance was used in Eisen, Spellman, Brown and Botstein (1998), and it has been our experience that it is a sensible choice in many applications. The ordering of clusters at each level is based on distances (with respect to the chosen metric) between the medoids of the clusters. Within clusters, the elements can be ordered based on distance to their own medoid or to the neighboring medoid.

By combining the strengths of two celebrated approaches to clustering, partitioning and agglomerative methods, HOPACH is a flexible, accurate algorithm for finding patterns in data. van der Laan and Pollard discuss discuss simulations and data analyses which show that HOPACH-PAM is better able to identify patterns in data than either a single partitioning or agglomerative clustering. For example, in one simulation with seven gene clusters plus unrelated noisy genes, HOPACH-PAM produces a clustering result with seven clusters and an average silhouette of 0.30, whereas simply applying PAM to the data and maximizing silhouette to choose the number of clusters produces a clustering with only six clusters and an average silhouette of 0.09.

HOPACH can be applied along with pre- and post-screens if, for example, we wish to only consider significantly differently expressed genes or only genes which are strongly correlated with other genes in their cluster. By working with all genes, however, we are able to avoid the problem of accidentally removing genes of interest.

Clustering all genes

We applied HOPACH-PAM with the cosine-angle distance metric to the combined data set containing all 3500 genes. The clustering of this data set is of interest since it will contain groups of genes significantly over- and under-expressed in melanoma relative to breast cancer in addition to genes with similar mean expression in both cancers. We identified seven clusters, five under-expressed and

two over-expressed. Two of the clusters within the under-expressed group had medoids which were not two-fold under-expressed (average across cell lines), and these clusters contained many genes not significantly differently expressed. The third under-expressed cluster contains a core group of genes which are very highly correlated with each other. Panel A of Figure 5 contains the distance matrix with genes ordered relative to distance from their own medoid within each cluster. We picture only a random subset of 1000 of the 3500 genes, because the image containing fewer genes has identical structure and is easier to work with.

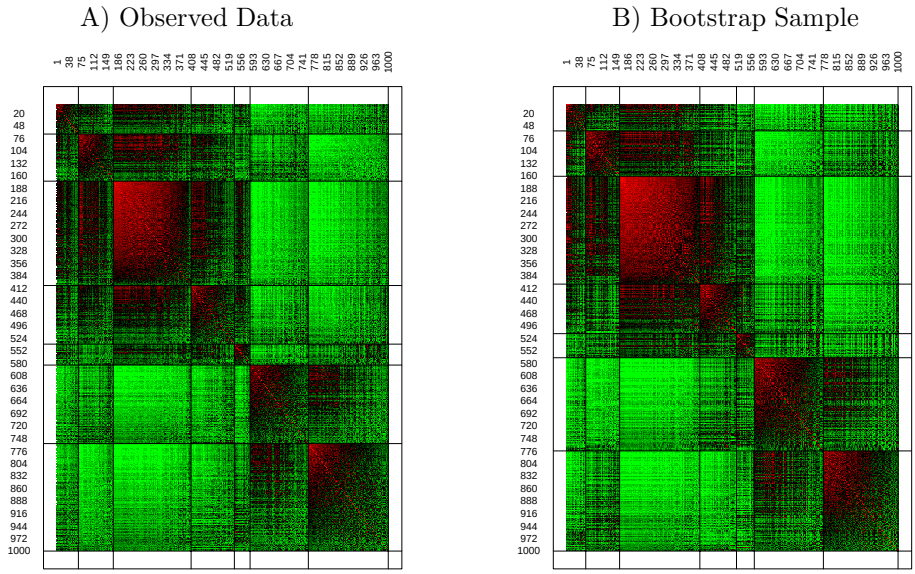

Figure 5. Ordered distance matrices from clustering all genes in NCI combined data set (random subset pictured). Panel A contains the observed data distance matrix. Panel B contains a non-parametric bootstrap sample distance matrix. Red corresponds with smallest and green with largest distance. Lines indicate cluster boundaries.

We ran the non-parametric bootstrap with fixed medoids in order to estimate the variability of gene cluster membership. The cluster probability plot corresponding with the ordered distance matrix is pictured in Figure 6. We can see that some clusters are more stable than others (e.g., Cluster 3, the strong under-expressed cluster) and that there are pairs of clusters (e.g., Clusters 6 and 7, the over-expressed clusters) which are similar in the sense that they often exchange genes between bootstrap samples. In order to estimate the variability of the overall clustering pattern (including the selection of medoids), we also ran the non-parametric bootstrap without fixing the medoids. For each bootstrap

sample, we applied HOPACH-PAM. For this data set, the number and size of clusters was very stable despite the fact that the exact choice of medoid did vary considerably from sample to sample. When it is possible to infer a correspondence between original and bootstrap medoids (i.e., when cluster sizes or profiles vary greatly), then summary measures relating to specific clusters can still be calculated. In general, we can still visualize the overall clustering patterns. The reordered distance matrix for one random bootstrap sample is pictured in Panel B of Figure 5. We see that the overall clustering pattern is very similar to that found in the original data (Panel A). One could visualize many such distance matrices or calculate summary measures of these in order to quantify the overall variability of the clustering procedure.

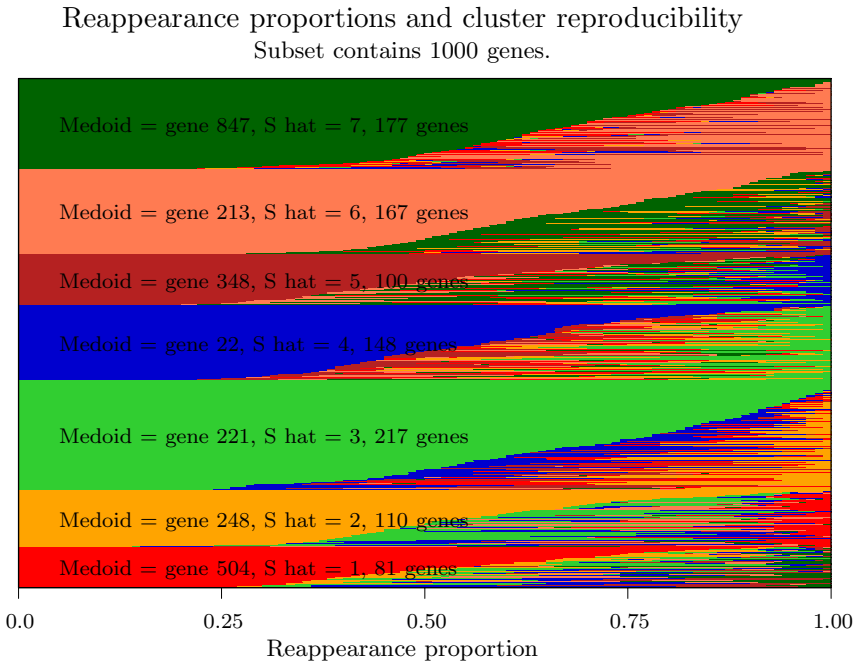

Figure 6. Reappearance probabilities from clustering all genes in NCI combined data set (random subset pictured).

Clustering differently expressed genes

It is also of interest to cluster only the differently expressed genes. HOPACH-PAM with cosine-angle distance identified four clusters, one large cluster containing genes under-expressed and three smaller clusters containing genes over-expressed in melanoma relative to breast cancer. We found that when the

non-differently expressed genes had been removed from the data set, the under-expressed genes were all strongly correlated with each other and negatively correlated with the over-expressed genes so that the reordered distance matrix had a clear block structure. Whereas the cluster of under-expressed genes is very homogeneous, the group of over-expressed genes consists of several distinct clusters. The bootstrap methodology was applied to this clustering result in a similar manner as for all genes.

Clustering the two cancers separately

As an alternative way to compare the breast and melanoma subpopulations, we applied HOPACH-PAM with cosine-angle distance to the data sets for each type of cancer separately. In both data sets, the genes were separated into over- and under-expressed in the initial level of the tree, with about two thirds of the genes being under-expressed in tumor relative to the pooled reference sample. In the next level of the tree, the over-expressed genes were split again into two smaller clusters. For breast cancer these two clusters contained about 600 and 300 genes each, whereas for melanoma most over-expressed genes were in a cluster together with the remainder in a small cluster of size 25. Comparing the reordered distance matrices for each of the two types of cancer, we can see these similarities and differences. In addition, for any gene of interest, we can compare where it appears in the ordering for each type of cancer and what other genes it appears near in a cluster. Differences in the biological mechanisms involved in the two types of tumors might be explained by differences in their clustering structures. For example, melanoma has stronger negative correlation between many of the over- and under-expressed genes, indicating that these groups of genes might be working together. The bootstrap methodology was applied to this clustering result in a similar manner as for all genes.

The method for identifying differently expressed genes could also be applied to the melanoma and breast cancer cell line data sets separately to find which genes were differently expressed relative to the reference sample in each data set. These lists could be compared or clustered separately.

4. Asymptotic Performance of the Methods and a Non-asymptotic Sample Sample Size Formula

The rules for subsetting and clustering genes used in this paper are functions of the empirical mean vectors and empirical covariance matrices of the two samples of size n_1 and n_2, respectively. For example, the clustering of genes is a function of a user-supplied distance matrix, while the euclidean distance $\|\vec{x}-\vec{y}\|$, correlation-distance $1-\langle\vec{x},\vec{y}\rangle/\|\vec{x}\|\,\|\vec{y}\|$ before and after centering at zero, and the absolute correlation distance are all functions of the empirical covariance matrix.

Therefore we can view subsetting and clustering methods as estimates $\widehat{\mathbf{S}}$ of an unknown parameter $\mathbf{S}(\mu_1, \Sigma_1, \mu_2, \Sigma_2)$.

The most important quality of the estimated subset rule to establish is its consistency in the relevant context of $n/\log(p) \to \infty$. Since the estimated subset is a function solely of the subpopulation sample moments, consistency of the sample moments, together with continuity of the subset rule, will imply consistency of $\widehat{\mathbf{S}}$. For a fixed number of genes p, the Law of Large Numbers guarantees that the subpopulation sample moments are consistent for the true moments, uniformly across all genes. However, in practice, the number of genes being studied continues to grow and to grow much more rapidly than sample size. We find it valuable to state conditions under which the estimated subset is consistent, even in the realistic setting where the arrays keep getting larger and larger.

By restricting the log ratio data to a compact interval, we can avoid making any other assumptions about the data-generating distribution. For this reason, we replace any log ratio $\log X_{d,j}$ $(d = 1, 2)$ that is farther than some fixed M, $0 < M < \infty$, from its mean $\mu_{d,j}$ with the value $\mu_{d,j} \pm M$, where the sample mean $\hat{\boldsymbol{\mu}}_d$ is used in place of $\boldsymbol{\mu}_d$ in a data analysis. We include truncation by the constant M as part of the transformation of the raw relative expression data. In van der Laan and Bryan (2001) it is shown that if the number of genes $p = p(n_d)$ in sample d is such that $n_d/\log(p(n_d)) \to \infty$ as $n_d \to \infty$, then, $\max_j |\hat{\mu}_{d,j} - \mu_{d,j}| \to 0$ in probability and $\max_{ij} |\widehat{\Sigma}_{d,ij} - \Sigma_{d,ij}| \to 0$ in probability. Therefore, for a continuous subset rule $\mathbf{S}$, $P(\mathcal{S} = \widehat{\mathbf{S}}_n) \to 1$ in probability. The proof of this theorem is a direct consequence of Bernstein's Inequality (see van der Vaart and Wellner (1996), page 102) and is given in van der Laan and Bryan (2001), along with a precise definition of the required continuity.

The same Bernstein Inequality argument used above also leads to a sample size formula in the more concrete setting of a fixed value of the number of genes p. For the d-th sample $(d = 1, 2)$, we define n_d^* with the following formula:

$$n_d^*(p, \epsilon_d, \delta_d, M, \sigma_d^2) = \frac{1}{c}(\log p + \log \frac{2}{\delta_d}),$$
$$\text{where } c = c(\epsilon_d, \sigma_d^2, M) = \frac{\epsilon_d^2}{2\sigma_d^2 + 2M\epsilon_d/3}. \tag{1}$$

In the above, σ_d^2 is an upper bound of $\max_j \sigma_{d,j}^2$ and δ_d is a user-specified value between 0 and 1 that can be thought of as 1 minus the "power". If $n_d > n_d^*$, then $P(\max_j |\hat{\mu}_{d,j} - \mu_{d,j}| > \epsilon_d) < \delta_d$. Similarly, if $n_d > n_d^*(p^2, \epsilon_d, \delta_d, M^2, \sigma_{d,\Sigma}^2)$, where $\sigma_{d,\Sigma}^2$ is an upper bound of the variance of $Y_{d,j}Y_{d,k}$, then $P(\max_{ij} |\widehat{\Sigma}_{d,ij} - \Sigma_{d,ij}| > \epsilon_d) < \delta_d$. The constant ϵ_d is the maximum tolerable distance between sample means and true means and the sample size formula guarantees subpopulation sample means within this distance from the truth with probability $1 - \delta_d$.

To see how one might use this formula in practice, consider the "extemeley false positive" genes. We are concerned when such a gene appears in $\widehat{\mathbf{S}}$ and want to know how often this might happen. For example, in the case with one set of observations, suppose the target subset contains genes with absolute mean expression greater than 1 (that is, more than $2^1 = 2-$fold differential expression) and we want to choose a sample size that guarantees genes with absolute mean expression less than 0.14 (that is, less than $2^{0.14} \approx 1.1-$fold differential expression) rarely appear in $\widehat{\mathbf{S}}_n$. In this case, the maximum tolerable distance is $\epsilon = 1-0.14 = 0.86$. If all sample means are within ϵ of their respective true means with probability $1-\delta$, then the probability of an extremely false positive is bounded above by δ.

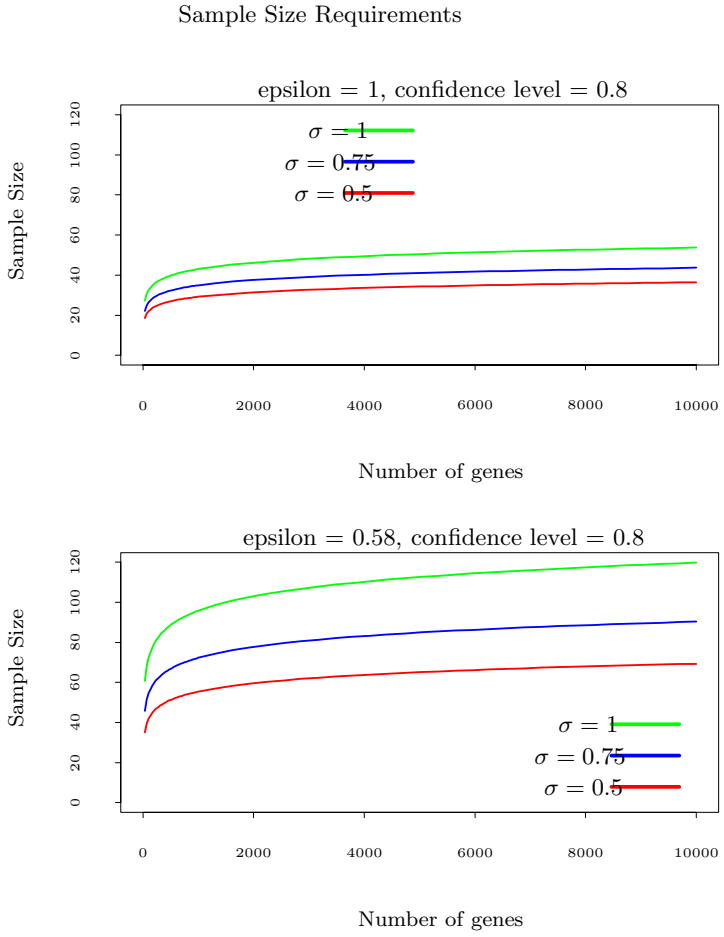

Figure 7. Sample size requirements for different situations.

Figure 7 illustrates the implications of this sample size formula for several

realistic scenarios. In the top panel we have set $\epsilon = 1$ and in the bottom $\epsilon = 0.58$. Decreasing the tolerable distance ϵ, holding all other constants fixed, increases the required sample size. The noise levels implied by the values of σ in the range [0.5, 1.0] are typical for the data sets we have seen. We note that the sample size formula is actually quite conservative. It makes no assumptions about the correlation between the p genes and, when there is a significant amount of correlation, the true dimension of the problem can be much smaller than p. In practice, given the highly correlated microarray data, we see that the sample size formula produces extremely false positive rates much lower than the nominal rate of δ.

It is of interest to see that the effect of the number of genes on this sample size formula (and the truly needed sample size) is minimal. In other words, if one needs a certain sample size for 10 genes, then adding 50 subjects to your sample will guarantee the same uniform precision based on 100000 genes. It teaches us that achievable sample sizes will allow complete trust in *each* of the elements of the observed mean vectors and distance matrices, which will become essential if one is interested in selecting association pathways between genes. For detecting global clustering structures as carried out in this paper a uniform precision on the distance matrix is not needed. van der Laan and Bryan (2001) also provide a corresponding simultaneous finite sample confidence band for μ_j, $j = 1, \ldots, p$. The authors prove asymptotic validity of the parametric bootstrap when treating the standardized empirical mean and covariance as random elements of an infinite dimensional weighted-euclidean Hilbert space. We remark that their proofs can be immediately applied to establish these asymptotic validity results for the nonparametric bootstrap.

Finally, if subset rules are based on non-linear statistics such as quantiles (e.g., medians), then one can base the consistency theorem and sample size formula on the first order linear expansion of these statistics. That is, for the jth statistic $\hat{\theta}_j$ (j indexes a gene-specific quantile or a gene-pair specific distance) we will have under appropriate assumptions that $\hat{\theta}_j - \theta_j \approx 1/n \sum_{i=}^{n} IC(X_i)$, where IC is the influence curve. In addition, one needs to control the second order terms uniformly in j, which means that one needs to make an assumption uniformly bounding away j-specific singularities.

Acknowledgements

This research has been supported by NIAID grant 1R01 AI46182-01 and Life Sciences Informatics Program Grant L98-10050 with industrial sponsor Chiron Corporation, Emeryville, CA (http://www.chiron.com).

References

Dudoit, S., Fridlyand, J. and Speed, T. (2000). Comparison of discrimination methods for the classification of tumors using gene expression data. Technical Report 576, Statistics Department, University of California, Berkeley.

Eisen, M. B., Spellman, P. T., Brown, P. O. and Botstein, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci.* **95**, 14863-14868.

The Chipping Forcast (1999). *Nature Genetics* **21** (1, suppl.).

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., Mesirov, J. P., Coller, H., Loh, M. L., Downing, J. R., Caligiuri, M. A., Bloomfield, C. D. and Lander, E. S. (1999). Molecular classification of Cancer: Class discovery and class prediction by gene expression monitoring. *Science* **286**, 531-537.

Hastie, T., Tibshirani, R., Eisen, M. B., Alizadeh, A., Levy, R., Staudt, L., Chan, W., Botstein, D. and Brown, P. O. (2000). Gene shaving as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology* **1**.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis.* John Wiley, New York.

Marshall, E. (1999). Do-it-yourself gene watching. *Science* **286**, 444-447.

Perou, C. M., Sørlie, T., Eisen, M. B., van de Rijn, M. Jeffrey, S. S., Rees, C. A., Pollack, J. R., Ross, D. T., Johnsen, H., Akslen, L. A., Fluge, Ø., Pergamenschikov, A., Williams, C., Zhu, S. X., Lønning, P. E., Børresen-Dale, A. L., Brown, P. O. and Botstein, D. (2000). Molecular portraits of human breast tumors. *Nature* **406**, 747-752.

Pollard, K. S. and van der Laan, M. J. (2001). Statistical inference for two-way clustering of gene expression data. Technical Report 96, Group in Biostatistics, University of California, Berkeley, to appear.

Ross, D. T., Scherf, U., Eisen, M. B., Perou, C. M., Rees, C., Spellman, P., Iyer, V., Jeffrey, S. S., Van de Rijn, M., Waltham, M. Pergamenschikov, A., Lee, J. C. F., Lashkari, D., Shalon, D., Myers, T. G., Weinstein, J. N., Botstein, D. and Brown, P. O. (2000). Systematic variation in gene expression patterns in human cancer cell lines. *Nature Genetics* **24**, 227-235.

van der Laan, M. J. and Bryan, J. F. (2001). Gene Expression Analysis with the Parametric Bootstrap. *Biostatistics* **2**, 1-17.

van der Laan, M. J. and Pollard, K. S. (2001). Hybrid clustering of gene expression data with visualization and the bootstrap. Technical Report 93, Group in Biostatistics, University of California, Berkeley, submitted.

van der Vaart, A. and Wellner, J. A. (1996). *Weak Convergence and Empirical Processes.* Springer, New York.

Biotechnology Laboratory and Department of Statistics, University of British Columbia, 333-6356 Agricultural Road, Vancouver, BC V6T 1Z2, Canada.

E-mail: jenny@stat.ubc.ca

School of Public Health, Division of Biostatistics, University of California, Berkeley, CA 94720-7360, U.S.A.

E-mail: kpollard@stat.berkeley.edu

School of Public Health, Division of Biostatistics, University of California, Berkeley, CA 94720-7360, U.S.A.

E-mail: laan@stat.berkeley.edu

(Received March 2001; accepted October 2001)