

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. A typical 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. Consider, for example, a population of cancer patients from which we take a random sample of n patients, each of whom contributes p gene expression measurements. For microarrays, each measurement is a ratio, calculated from the intensities of two fluorescently labeled mRNA 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. Data preprocessing may include background subtraction, combining data from replicated spots representing the same cDNA sequence, normalization, log transformation, and truncation.

Given data from such an experiment, researchers are 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. For example, healthy and cancerous cells can be compared within subjects in order to learn which genes tend to be differentially expressed together in the diseased cells; regulation of such genes could produce effective cancer treatment and/or prophylaxis [4,5,12,14]. In addition to identifying interesting clusters of genes, researchers often want to find subgroups of samples which share a common gene expression profile. Examples of such studies include classifying sixty human cancer cell lines [16], distinguishing two different human acute leukemias [9], dissecting and classifying breast cancer tumors [15], and classifying sub-types of B-cell lymphoma [1] and cutaneous malignant melanoma [2].

Most gene expression research to date has focused on either the gene problem or the sample problem, or possibly on the two problems as separate tasks. Many researchers have performed two-way clustering by applying algorithms to both genes and samples separately and possibly reordering the rows and columns of the data matrix according to the clustering results. Tibshirani *et al.* [17], for example, illustrate several methods for two-way visualization of a reordered data matrix based on separately clustering genes and samples. They also propose applications of block clustering and principal components analysis to simultaneously cluster genes and samples. Their gene shaving methodology addresses the problem that different subsets of genes might cluster samples in different ways.

We want to emphasize the importance of this last issue. Consider, for example, that different gene clusters represent different biological mechanisms or states, so that there may be clusters of genes which are very good for distinguishing different types of samples. In addition to an overall clustering label, sub-groups of samples could be more accurately characterized by their expression pattern for each of these gene clusters. In this way, we might find that in an experiment consisting of three types of cancer, several gene clusters show similar expression across all three types (generic cancer genes), one gene cluster shows two types of cancer clustering together, and another gene cluster shows a different two types of cancer clustering together. It is possible that clustering the samples using all genes might not have revealed the three types of cancer, whereas the characterization based on the patient clustering within separate gene clusters would make the distinction clear. Similarly, first clustering samples and then genes within sample clusters could offer more insight than simply clustering genes. For example, in a heterogeneous sample of patients, clustering patterns seen in individual patient groups might disappear when averaged across groups or the profile of the more numerous patient group might dominate. Concern over failing to identify these more subtle and complicated patterns in gene expression data has motivated the development of a general statistical framework for simultaneous clustering.

2 Background

Approaches to gene expression data analysis rely heavily on results from cluster analysis (e.g.: k-means, self-organizing maps and trees), supervised learning (e.g.: recursive partitioning), and classification and regression trees (CART). We recommend the clustering algorithm Partitioning Around Medoids (PAM) [11], because it is nonparametric and more robust to outliers than many other methods. PAM takes a dissimilarity matrix (based on any distance metric of interest) as input. For a given number of clusters K , PAM selects K potential medoids from the observed data, identifies for all other elements the distance to the closest of these potential medoids, and minimizes over the vector of K potential medoids the sum of these distances over all elements. The solution to this minimization problem is the vector of K medoids, each identifying a cluster defined as those elements which are closer to that medoid than to any of the other $K - 1$ medoids. Because of how they are selected, the medoids are more stable than the mean vectors in k-means. The statistical framework we present can be applied with any choice of clustering algorithm, but we will use PAM to illustrate the methodology because of its nice properties.

All exploratory techniques are capable of identifying interesting patterns in data, but they do not inherently lend themselves to statistical inference. It is

necessary to define important statistical notions such as parameter, parameter estimate, consistency, and confidence. The ability to assess reliability in an experiment is particularly crucial with the high dimensional data structures and relatively small samples presented by gene-expression experiments. In the biotechnology industry, for example, where thousands of potential drug targets are screened simultaneously, it is imperative to have a method to determine the significance of gene clusters found in a single microarray experiment.

Others have noted this need for statistical rigor in gene expression data analysis [10,13]. Van der Laan and Bryan [18], present a general statistical framework for clustering genes using a deterministic subset rule applied to (μ, Σ) , the mean and covariance of the gene expression distribution. A typical subset rule will draw on “screens” and “labellers”. A screen is used to eliminate certain genes from the subset. A labeller will apply labels, such as the output of a clustering routine. Meaningful analyses can be done with various combinations of screens and labellers or even with a screen or labeller alone. For example, one might apply PAM to all genes at least two-fold differentially expressed. The target subset $S(\mu, \Sigma)$ is the subset (with cluster labels) the subset rule would select if the true data generating distribution were known, and it is estimated by the observed sample subset $S(\hat{\mu}_n, \hat{\Sigma}_n)$, where the empirical mean and covariance are substituted for the true parameters. Most currently employed clustering methods fit into this framework, since they need only be deterministic functions of the empirical mean and covariance $\hat{\mu}_n, \hat{\Sigma}_n$. The authors also provide measures and graphs of gene cluster stability based on the parametric bootstrap using a truncated multivariate normal distribution $N(\hat{\mu}_n, \hat{\Sigma}_n)$. Finally, they establish consistency under $\frac{n}{\log(p)} \rightarrow \infty$ of:

- (1) $\hat{\mu}_n, \hat{\Sigma}_n$ and hence smooth functions $S(\hat{\mu}_n, \hat{\Sigma}_n)$.
- (2) the parametric bootstrap for the limiting distribution of $\sqrt{n}(\hat{\mu}_n - \mu, \hat{\Sigma}_n - \Sigma)$ and simple convergence of the bootstrap subset to the true subset.

3 Simultaneous Clustering Parameter

In this paper we propose a generalization of the method of van der Laan and Bryan [18] such that the subset rule may include a multi-stage clustering method which involves simultaneous clustering of genes and samples. In order to be concrete, we will refer to the samples as patients, but the methodology applies to any i.i.d. experimental units. Define a simultaneous clustering parameter as a function of the true data generating distribution which is a composition of a mapping involving clustering of patients and a mapping involving clustering of genes. An estimate of the simultaneous clustering parameter is obtained by applying this function to the empirical distribution. This formal framework allows us to assess classical properties of the clustering

method such as consistency (in the novel context of $n \ll p$) and also allows us to study statistical inference regarding the clustering parameter.

3.1 Clustering Patients

Given k_1 and an m -variate distribution P , let $\Phi_1(P) = (P_j(P), p_j(P) : j = 1, \dots, k_1)$ be an algorithm that maps P into k_1 m -variate distributions P_1, \dots, P_{k_1} and corresponding mixing proportions p_1, \dots, p_{k_1} . This mapping $\Phi_1(P)$ represents clustering of patients.

For example, one could fit to P a mixture of k_1 m -variate normal distributions. In other words, let $f(x | (\mu_j, \Sigma_j, p_j), j = 1, \dots, k_1)$ be the density of a mixture $\sum_j p_j N(\mu_j, \Sigma_j)$ of k_1 multivariate normal distributions. Now, we could define $\Phi_1(P) = \max^{-1} \int \log(f(X | (\mu_j, \Sigma_j, p_j), j = 1, \dots, k_1)) dP(x)$ as the distribution (i.e.: parameters) which maximizes the log-likelihood, where the maximum is taken over all mixtures of multivariate normal distributions. In this case, the algorithm involves maximum likelihood estimation over a mixture of parametric families, which could be carried out with the EM-algorithm. This clustering methodology is carried out by Fraley and Raftery [7], and it is referred to as model based clustering.

The alternative to model based clustering is nonparametric clustering which is much less computer intensive. One particular method of nonparametric clustering is PAM. In this case, the easiest way to define $\Phi_1(P)$ is by simulation. In other words,

- (1) sample an infinitely large number (say N) observations from P ,
- (2) compute a $N \times N$ -distance matrix containing the pairwise distances for each pair of observations for some specified distance metric,
- (3) apply the clustering algorithm PAM to split the N observations into k_1 groups,
- (4) for each group j , report the proportions of observations p_j and the empirical cumulative distribution function P_j of these observations, $j = 1, \dots, k_1$.

If $N = \infty$, then the output of this simulation algorithm is a function $\Phi_1(P) = (P_j, p_j : j = 1, \dots, k_1)$.

3.2 Clustering Genes

Given k_2 and an m -variate distribution Q , let $\Phi_2(Q) = (m_j(Q), S_j(Q), G_j(Q) : j = 1, \dots, k_2)$ be an algorithm that maps Q into k_2 $m_j(Q)$ -variate subdistribu-

tions $G_1(Q), \dots, G_{k_2}(Q)$ of Q corresponding with subsets $S_j(Q)$ of $\{1, \dots, m\}$ of sizes m_j . This algorithm represents the clustering of m genes into k_2 clusters S_j of genes of sizes m_j , $j = 1, \dots, k_2$. We might also extend the definition of S_j so that it includes not just the clusters, but also other output of a clustering algorithm such as probabilities that genes belong to a cluster (fuzzy membership).

As highlighted in van der Laan and Bryan [18], the finding of the actual clusters $s_j(Q)$ of genes, $j = 1, \dots, k_2$, can typically be based on the m -variate mean vector $\mu(Q)$ and the $m \times m$ -covariance matrix $\Sigma(Q)$ of Q . In that case, $s_j(Q) = s_j(\mu(Q), \Sigma(Q))$, $j = 1, \dots, k_2$. For example, $\Phi_2(Q)$ can be defined by applying PAM to a $m \times m$ -distance matrix (Euclidean, correlation, absolute correlation, etc.) calculated from the $m \times m$ -covariance matrix $\Sigma(Q)$.

3.3 Compositions

To summarize we have:

$$\begin{aligned}\Phi_1(P) &= (p_j(P), P_j(P) : j = 1, \dots, k_1) \\ \Phi_2(Q) &= (m_j(Q), S_j(Q), G_j(Q) : j = 1, \dots, k_2),\end{aligned}$$

where Φ_1 and Φ_2 are defined by what algorithm you want to use (e.g.: clustering with a specified distance metric). Note that Φ_1 splits the population into k_1 subpopulations so that applied to patients it will split the sample into k_1 subsamples, while Φ_2 splits the dimension into subdimensions. These mappings $\Phi_1(P)$ and $\Phi_2(Q)$, representing clustering of patients and genes, are the building blocks for simultaneous clustering parameters $\Phi(P)$.

Define the composition $\Phi_2 \circ \Phi_1(P)$ as a $k_1 \times k_2$ -matrix with (I, J) -th element being $(p_I(P), P_I(P), m_J(P_I(P)), S_J(P_I(P)), G_J(P_I(P)))$, i.e. it reports the I -subpopulation distribution of patients computed by $\Phi_1(P)$ and the corresponding J -th cluster of genes. Similarly, define the composition $\Phi_1 \circ \Phi_2(P)$ as a $k_2 \times k_1$ -matrix with (I, J) -element being $(m_I(P), S_I(P), G_I(P), p_J(G_I(P)), P_J(G_I(P)))$, i.e. it reports the I -th cluster of genes and clusters patients just based on the genes in this I -th cluster. Then, $\Phi_2 \circ \Phi_1(P)$ represents clustering genes within each cluster of patients and $\Phi_1 \circ \Phi_2(P)$ represents clustering patients within each cluster of genes. Stopping at this stage often reveals interesting patterns in the data.

One might find it of interest to iterate these compositions in order to design more “aggressive” algorithms. The mappings can be composed alternately. For example, one might 1) cluster genes, 2) within each cluster of genes cluster patients, and 3) within each cluster of patients cluster genes again. The result-

ing simultaneous clustering parameter $\Phi_2 \circ \Phi_1 \circ \Phi_2(P)$ reports the I -th cluster of genes based on all patient data, the J -th cluster of patients just based on I -th cluster of genes and the K -th cluster of genes just based on this J -th cluster of patients. It is natural to think of such a multi-way-array as a tree which is many levels deep so that to get to the (I, J, K) -th element of the tree, for example, one goes to branch I at the first node, branch J at the second node and branch K at the third node. One could also iterate one mapping repeatedly. Define

- (1) $\Phi_1 \circ \Phi_1(P)$ as a $k_1 \times k_1$ -matrix with (I, J) -th element being

$$(p_I(P), P_I(P), p_J(P_I(P)), P_J(P_I(P))),$$

i.e. it reports the I -th cluster of patients based on all data and the J -th cluster of patients just based on this I -th cluster of patients.

- (2) $\Phi_2 \circ \Phi_2(P)$ as a $k_2 \times k_2$ -matrix With (I, J) -th element being

$$(m_I(P), S_I(P), G_I(P), m_J(G_I(P)), S_J(G_I(P)), G_J(G_I(P))),$$

i.e. it reports the I -th cluster of genes based on all data and the J -th cluster of genes just based on this I -th cluster of genes.

Thus, clustering of samples (or genes) is defined within an earlier obtained cluster of samples (or genes). This is top-down (or divisive) hierarchical clustering. It is easy to see that repeatedly applying Φ_1 or Φ_2 results a final hierarchical tree structure of the type produced by Eisen's application of bottom-up (agglomerative) hierarchical clustering [6]. Note that contrary to his algorithm, we are not necessarily restricted to binary splits, making this approach more general. The block clustering method for gene expression data described in Tibshirani *et al.* [17] can also be expressed as a hierarchical iteration which includes both Φ_1 and Φ_2 . It is important to remember that applying such iterated algorithms to the actual data yields very aggressive search algorithms. We must keep in mind that as more compositions are taken, the mapping becomes less smooth so that there is a trade off between stability (i.e.: sample size needed) and an aggressive search (i.e.: finding patterns in the data).

Now, we can define a simultaneous clustering parameter $\theta = \Phi(P)$ as some (possibly iterative) composition of Φ_1, Φ_2 .

3.4 Summary Measures of the Simultaneous Clustering Parameter

It is useful to consider various summary measures of the simultaneous clustering parameter θ . Examples of such summary measures include:

- **Cluster Membership (Genes only):** probabilities (fuzzy clustering), assignments (hard clustering). Recall that patient cluster labels are not a parameter.
- **Cluster Size:** proportion in each cluster
- **Cluster Profile:** means, medoids, indicator that a gene cluster shows strong differential expression (uniform) in at least one of the patient clusters.
- **Cluster Strength:** diameter, separation, silhouettes, measures of how uniformly the genes behave across patients.

Many of these measures can be summarized together in a picture. For example, we can order the gene clusters within each cluster of patients, and then order patients and genes within clusters. Ordering can be based on distance between clusters, silhouette, or some dimension reducing projection (multidimensional scaling, principal components). For each cluster of patients, we might visualize (i) the reordered data matrix and (ii) the reordered gene-by-gene correlation (or more generally distance) matrix.

3.5 Estimation and Remarks Regarding Asymptotic Consistency

Let P_n be the empirical distribution of the data. We estimate $\theta = \Phi(P)$ with $\theta_n = \Phi(P_n)$. Note that this is a nonparametric estimate since it is not based on model assumptions. In particular, if $\Phi : (D_1, \|\cdot\|_1) \rightarrow (D_1, \|\cdot\|_2)$ is continuous w.r.t to a metric $\|\cdot\|$ for which $\|P_n - P\| \rightarrow 0$ in probability and p is fixed, then $\|\theta_n - \theta\|_1 \rightarrow 0$ in probability.

Since, in practice, the number of genes being studied p continues to grow and to grow much more rapidly than sample size n , it is of interest to establish consistency of summary measures of θ_n in the context that the number of genes $p = p(n)$ increases with n such that $n \rightarrow \infty$ and $n/\log(p(n)) \rightarrow \infty$. In general, simultaneous clustering parameters will be much less smooth than $S(\mu, \Sigma)$ from van der Laan and Bryan [16], and thus consistency, sample size and asymptotic validity of the bootstrap are issues we need to address. One benefit of having a composition is that we only need consistency of each mapping alone in order to show consistency of their composition. Since we already have consistency for gene clustering, it remains to look at patient clustering.

In this section, we will explain why in principal under appropriate regularity conditions we can extend the consistency and sample size proofs of van der Laan and Bryan [16] to the simultaneous clustering context. Let $\theta_j = \theta_j(P)$ be a real valued parameter of the data generating distribution P of the p -dimensional gene expression profiles X , where j indexes a large set of such parameters, say $j = 1, \dots, r(p)$. Here $r(p)$ is monotone function with bounded derivative in the number of genes p . In particular, we can view θ_j as one of

the many parameters of a simultaneous clustering $\Phi(P) = \Phi_1 \circ \Phi_2(P)$. For a fixed number of genes p and increasing sample size n one will typically have that the empirical estimate $\theta_{jn} = \theta_j(P_n)$ is asymptotically linear:

$$\theta_{jn} - \theta_j = \frac{1}{n} \sum_{i=1}^n IC_j(X_i) + r_{jn},$$

where $|r_{jn}| = o_P(1/\sqrt{n})$. Here $Y \rightarrow IC_j(X)$ is the influence curve of the estimator θ_{jn} and $E\{IC_j(X)\} = 0$. Let $\sigma_j^2 = \text{VAR}IC_j(X)$ denote the variance of the influence curve $IC_j(X)$.

Now we can state conditions under which the parameters θ_{jn} are uniformly consistent in j , even in the realistic setting where the arrays keep getting larger and larger. Let's now assume that (1) the log gene expressions are truncated, so that they are bounded from above by a universal constant M , (2) the influence curves IC_j are uniformly bounded in all possible X 's by a universal constant C , and (3) the second order term $\sqrt{n}r_{jn}$ converges to zero uniformly in j in probability. Condition 3 requires typically the same assumptions as condition 2. In other words, once one makes sure that denominators in the influence curve of the estimator θ_{jn} are uniformly bounded away from zero, one will often also be able to prove a uniform bound on the second order terms.

Let $\tilde{\theta}_{jn} \equiv \theta_j + \frac{1}{n} \sum_i IC_j(X_i)$ be the first order approximation of θ_{jn} . In van der Laan and Bryan [16] it is shown that if the number of genes $p = p(n)$ is such that $n/\log(p(n)) \rightarrow \infty$ as $n \rightarrow \infty$, then, as $n \rightarrow \infty$,

$$\max_j |\tilde{\theta}_{jn} - \theta_j| \rightarrow 0 \text{ in probability,}$$

so that, by condition 3,

$$\max_j |\theta_{jn} - \theta_j| \rightarrow 0 \text{ in probability.}$$

The same Bernstein's Inequality argument as used in van der Laan and Bryan [16] leads to a sample size formula based on the first order approximation $\tilde{\theta}_{jn}$ of θ_{jn} in the more concrete setting of a fixed value of the number of genes p . Define n^* with the following formula:

$$n^*(p, \epsilon, \delta, C, \sigma^2) = \frac{1}{c} (\log p + \log \frac{2}{\delta}),$$

where $c = c(\epsilon, \sigma^2, C) = \frac{\epsilon^2}{2\sigma^2 + 2C\epsilon/3}$. In the above, $\sigma^2 = \max_j \sigma_j^2$ and δ is a

user-specified value between 0 and 1 that can be thought of as 1 minus the “power”. If $n > n^*$, then

$$P\left(\max_j |\tilde{\theta}_{jn} - \theta_j| > \epsilon\right) < \delta.$$

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 very minimal; in other words, if one needs a certain sample size for 10 genes, then adding 50 subjects to the sample will guarantee the same uniform precision based on 100000 genes. This teaches us that achievable sample sizes will allow complete trust in *each* of the elements of the observed estimates θ_{jn} , which will become essential if one is interested in selecting association pathways between genes.

4 Statistical Inference with the Bootstrap

Though $\theta = \Phi(P)$ generates a large set of methods for finding clustering patterns in the true data-generating distribution, once applied to empirical data P_n , it is likely to find patterns due to noise. To deal with this issue, one needs methods for assessing the variability of θ_n and, in particular, assessing the variability of the important summary measures of θ_n . One also needs to be able to test if certain components of θ_n are significantly different from the value of these components in a specified null-experiment.

To assess the variability of the estimator θ_n we propose to use the bootstrap. The idea of the bootstrap method is to estimate the true data generating distribution P with some estimate \mathbf{P}_n and estimate the distribution of θ_n with the distribution of $\theta_n^\# = \Phi(\mathbf{P}_n^\#)$, where $\mathbf{P}_n^\#$ is the empirical distribution based on an i.i.d. bootstrap sample (i.e.: a sample of n i.i.d. observations from \mathbf{P}_n). The distribution of $\theta_n^\#$ is obtained by applying $\theta = \Phi(\cdot)$ to $\mathbf{P}_n^\#$ from each of B bootstrap samples, keeping track of parameters of interest. The distribution of a parameter is approximated by its empirical distribution over the B samples.

There are several common methods for generating bootstrap samples.

- **Nonparametric:** Resample n columns from X with replacement.
- **Parametric:** Fit a model (e.g.: multivariate normal, mixture of multivariate normals) and generate observations from the fitted distribution.
- **Convex pseudo-data:** For $d \in \{0, 0.5\}$, choose $\epsilon \in \{0, d\}$. Then use ϵ to form new samples as convex combinations of pairs of randomly sampled columns of X . This is a smoothed version of the nonparametric bootstrap proposed by Breiman [3].

The nonparametric bootstrap has the advantage of being computationally much easier than the parametric bootstrap. In addition, the nonparametric bootstrap avoids distributional assumptions about the parameter of interest, whereas the estimation of the distribution of $\sqrt{n}(\Sigma_n - \Sigma)$ using the parametric bootstrap is only consistent under the model assumption. There is reason to believe, however, that the parametric bootstrap might perform better in the gene-expression context (where the number of observations n is typically very small relative to the dimension p), because the empirical distribution \mathbf{P}_n (i.e. nonparametric bootstrap) might be an inappropriate estimate of P .

5 Simulations to Assess the Bootstrap

The performance of the bootstrap is measured by how well the distribution of $\theta_n^\#$ approximates the distribution of θ_n . It is clear that this performance is mainly dependent on how close \mathbf{P}_n is to P . Our initial feeling was that in this setting, the empirical distribution P_n (i.e. nonparametric bootstrap) might be an inappropriate estimate of P . There is reason to believe, that the parametric bootstrap might perform better in the gene-expression context, where the number of observations n is typically very small relative to the dimension p (number of genes). Another fact of interest is that the nonparametric bootstrap is known to be inconsistent in various low-dimensional examples, while the parametric bootstrap is consistent under minimal additional assumptions given that the parametric model is correct [8]. The nonparametric bootstrap has the advantage, however, of being computationally much easier than the parametric bootstrap. In addition, the nonparametric bootstrap avoids distributional assumptions about the parameter of interest, whereas the estimation of the distribution of $\sqrt{n}(\Sigma_n - \Sigma)$ using the parametric bootstrap is only consistent under the model assumption.

With these ideas in mind, we conducted a set of simulation studies to assess the asymptotic validity of the nonparametric, convex, and parametric bootstraps for estimating the distribution of a simultaneous clustering parameter. Since θ can take many forms but is always a composition of the mappings Φ_1 and Φ_2 , we designed the simulations to look at Φ_1 and Φ_2 separately. In all of the simulations, we used $p = 3000$ genes and $n = 40$ samples. These choices reflect typical dimensions of the data matrix X (possibly after prescreening) as seen in commercial and academic settings. In order to investigate the effect of asymptotics on our results, we repeated all the simulations using $n = 250$ samples. We also repeated the Simulation 2 with $n = 150$ samples and $p = 12$ genes in order to look at the relative dependence of the asymptotics of patient clustering on the number of genes and the number of samples. All simulations and data analyses were carried out on a Dell Precision Workstation 620 with dual 1GHz processors and 2G of RAM using the statistical programming

languages R and Splus.

5.1 *Simulation 1: Multivariate Normal Data (with diagonal covariance)*

Simulation 1 investigates Φ_2 for gene clustering. The true data generating distribution was chosen to be a multivariate normal with diagonal covariance matrix so that the genes were uncorrelated. For simplicity, a fourth of the genes was generated from each of four distributions: $N(0.5, 0.25)$, $N(-0.5, 0.5)$, $N(1, 1)$, $N(-1, 0.75)$. The summary measures of interest were selected to be the 0.9 quantile of the maximum absolute difference in the mean vector, median vector, and correlation matrix. These measures give a good indication of how far a distribution is from the truth.

In order to define the “true” values of the summary measures, a large number N draws from the true distribution were compared to the known mean, median and correlation. Results were compared for $N = 100, 1000, 10000$ and showed little dependence on N so that $N = 100$ was deemed sufficient. Next, a single draw from the true distribution was identified as the “observed” data and the three types of bootstrap were performed with convex repeated for $d = 0.1, 0.3, 0.5$. In each case, $B = 100$ bootstrap samples were generated from which the 0.9 quantiles were calculated. In order to investigate the variability of these measures, we repeated each simulation twenty times with $n = 40$ samples and $p = 3000$ genes, obtaining twenty sets of 0.9 quantiles. From these, we calculated a mean and standard deviation. The coefficient of variation was on the order of 2.5% for the mean, 3.0% for the median and 1.25% for the correlation. These values were sufficiently small that we chose to use the results from just one simulation of $B = 100$ bootstrap samples in each case.

Table 1 shows the results of Simulation 1. We found that the bootstrap is good at $n = 250$ and a little conservative at $n = 40$. At both sample sizes, the bootstrap performed poorly for the median, which is a known result. It is interesting to note that in contrast to our hypothesis, the nonparametric bootstrap actually performed well relative to the convex and parametric bootstrap. We had expected the convex bootstrap, a smoothed version of the nonparametric, to perform consistently better than the nonparametric. Instead, we found that the convex was more biased for the mean than the nonparametric, performing best when d was smallest ($d = 0$ is equivalent to nonparametric).

This simulation suggests that the nonparametric and parametric bootstraps can be used to assess the variability of summary measures of gene clustering (see also van der Laan and Bryan [18]). Since estimated variability in the means is quite accurate and estimated variability in the correlation is accurate at $n = 250$ and conservative at $n = 40$, then we should be able to assess the variability

of subset rules of the form $S(\mu, \Sigma)$ accurately (or at least conservatively) for reasonable sample sizes.

5.2 *Simulation 2: Mixture of 2 Multivariate Normals (with diagonal covariance)*

Simulation 2 investigates Φ_1 for patient clustering. The true data generating distribution was chosen to be a mixture of two multivariate normals with identical, diagonal covariance matrices so that the genes were uncorrelated. The difference between the two component distributions in the mixture, then, was due solely to the difference in their mean vectors. Again, a fourth of the genes for each component distribution had one of four means, so that

$$(\mu_1, \mu_2) \in \{(0.5, -0.5), (0.25, -0.25), (-0.5, 0.5), (-0.25, 0.25)\}.$$

All variances were 2 and the mixing proportion was $q = 0.3$. Data was simulated from a mixture such as this by first selecting one of the two component distributions (with probability q of choosing the first) and then drawing a sample from that distribution. The parameter values for this simulation were selected so that there was sufficient overlap between the two component distributions in the mixture to occasionally confuse a clustering algorithm attempting to identify the source distribution of a randomly drawn sample.

The “true” values of summary measures were determined by comparing $N = 100$ draws from the true distribution to the mean value across the N samples. Next, a single draw from the true distribution was identified as the “observed” data and the three types of bootstrap were performed. In each case, $B = 100$ bootstrap samples were generated from which summary measures were computed in the same way.

In contrast to the gene clustering problem, where all genes appear in every sampled vector in a bootstrap sample, the original subjects are not all represented in a bootstrap sample. Indeed, in the convex bootstrap samples are now weighted averages over pairs of subjects, and in the parametric bootstrap all correspondence with original subjects is lost. We can not consider summary measures of θ that involve subject labels, since these are not parameters. We can, however, keep track of summary measures which pertain to clusters of subjects. These summary measures can be separated into those which depend on the cluster label (e.g.: cluster medoids or mean vectors) and those which do not (e.g.: features of the smallest cluster, differences between clusters). The former type of output requires supervised clustering. A supervised clustering procedure is one in which known clusters (e.g.: those found in the “observed” data set) are reused so that the subjects in each bootstrap sample are as-

signed to the closest of these clusters. An unsupervised clustering procedure allows new clusters to be identified in each bootstrap sample. We applied both supervised and unsupervised clustering as outlined below.

Supervised Clustering:

- For each subject in a bootstrap sample, compute the distance to each of the known mean vectors.
- Assign that subject to the closest cluster.
- Examples of summary measures: mixing proportion, cluster means.

Unsupervised Clustering:

- Apply the PAM algorithm ($k = 2$) to each bootstrap sample.
- The clusters do not necessarily correspond to the original clusters.
- Examples of summary measures: distance between medoids, diameter of the smaller cluster, average silhouette.

In a nonparametric unsupervised bootstrap, the cluster medoids will most likely be different from those from the original sample, and in a convex or parametric unsupervised bootstrap, the medoids can not be identical to those from the original sample. There are nonetheless several ways to infer a correspondence between the bootstrap and original clusters. If the clusters differ greatly in size and are well separated, as in the data analysis section, we may be able to infer a correspondence directly. Otherwise, we could order the clusters from the bootstrap sample with respect to distance relative to the original cluster medoids. Each bootstrap cluster would then “correspond” with the closest original medoid. In the case of a tie, the closer bootstrap cluster could be assigned to the original medoid and the other bootstrap cluster to the medoid next closest to it. Another approach to inferring a correspondence between each set of bootstrap clusters and the original clusters, is to align the clusters by examining the matrix of pair-wise distances between all bootstrap and original clusters and consecutively matching the closest pairs. The distance between clusters could be based on the distance between medoids or a measure of the overlap in membership, such as $(A \cap B)/(A \cup B)$ or $P(A|B)/2 + P(B|A)/2$.

Table 2 shows the results from supervised clustering. For each of the summary measures, the bootstrap estimates differed from the estimated true values. This difference was greatest for the convex bootstrap and reduced in magnitude for all three types of bootstrap with increasing n . These differences could be due to both bias and variance. In practice, the bootstrap is used to estimate the variance of an estimate, not its bias. So, we are more interested in whether the bootstraps correctly estimate this variance than in whether the bootstrap point estimates of summary measures of θ_n are biased for fixed n . For example,

the estimated standard errors for \hat{q} were quite good. Note that we report $\sqrt{n}\hat{\sigma}$ rather than the standard error of the bootstrap samples ($\hat{\sigma}$), since it is the quantity which should be used to evaluate the asymptotic consistency of the bootstrap. The bias in the centering of confidence intervals for \hat{q} and in the estimated mean vectors (at $n = 40$ versus $n = 250$) indicates that for patient clustering, larger sample sizes might be needed for unbiased point estimates. The results for $n = 150, p = 12$ are similar to those for $p = 3000$, indicating, as we have seen for gene clustering, that the asymptotics of patient clustering are driven more by n than by p . When additional genes are drawn from the same distribution, as in this simulation, having more genes actually provides more information about this distribution so that the high dimension of gene expression data might in fact be useful for estimation.

Table 3 shows the results from unsupervised clustering. Again, we see a bias in the bootstrap point estimates so that the estimated 95% CI's are not correctly centered. The parametric bootstrap did quite well, however, at estimating the width of the CI's (i.e.: $\hat{\sigma}$) even for $n = 40$. The nonparametric and convex bootstraps were conservative. Results for $n = 150, 250$ indicate that the nonparametric and parametric bootstraps are asymptotically valid for estimating the variability of summary measures of patient clustering. As seen in Simulation 1, the convex bootstrap performed relatively poorly. One possible explanation, in this case, might be that averaging two subjects characteristic of each of the component distributions creates a new "subject" which belongs equally to each underlying subpopulation and hence is more difficult to cluster.

We noted two interesting facts about the average silhouette. First, we found that average silhouette is a very stable parameter, which makes sense in light of the fact that it is an average over many elements. Secondly, we noticed that the parametric bootstrap may be optimistic about patient clustering. Consequently, parametric bootstrap confidence intervals for the average silhouette were of correct width but particularly biased (especially at $n = 40$), and this bias was in favor of stronger evidence of clustering than was seen in the observed data. We must remember, however, that since the silhouette is so stable, its variance is close to zero for relatively small sample sizes. Therefore, one can not expect that bootstrap confidence intervals will precisely cover the true confidence intervals. In practice, we would use the observed value for an estimate and perform the bootstrap only in order to estimate the variance of this estimate. Averaging across bootstrap samples to obtain a new estimate is called "bagging" and in this case does not seem to improve estimation.

5.3 Conclusions

Overall, these simulations indicate that larger sample sizes may be needed to avoid bias in estimates of the less smooth summary measures of patient clustering compared to the smoother summary measures of gene clustering. Nonetheless, the asymptotic validity of the nonparametric and parametric bootstraps for estimating the variability of these summary measures is clear in the results for $n = 250$. This is the true purpose of the bootstrap. Surprisingly, the nonparametric bootstrap performed relatively well compared to the parametric, although it was very conservative at $n = 40$ for estimating the variance of summary measures in the unsupervised clustering simulation. So, this may be a case where the extra computational effort of the parametric bootstrap would be worthwhile. The convex bootstrap had inconsistent performance and can not be recommended uniformly as an improvement to the nonparametric bootstrap. The results for $n = 150, p = 12$ indicate that the asymptotics of patient clustering, like gene clustering, are driven by the number of samples rather than the number of genes.

6 Data Analysis

In order to demonstrate the methodology presented in this paper, we have applied simultaneous clustering using PAM with a Euclidean distance metric to the publicly available data set published in Golub *et al.* [9]. Affymetrix GeneChips were measured for each of 38 leukemia patients, 27 with acute myeloid leukemia (AML) and 11 with acute lymphoblastic leukemia (ALL). Good gene expression data was available for 5925 genes. We truncated the expression measurements below by 100 and above by 16000. For computational ease, we reduced the dimension of the data set by selecting a subset of 2000 genes with greatest across patient variance. We would not necessarily recommend this prescreening procedure in general, since we have seen in other data sets that "interesting" genes (even when the goal is to cluster patients) are not necessarily always those with highest variance. Since our goal here was to demonstrate a methodology on a familiar data set, removing some potentially interesting genes was not so problematic. Prescreening based on expression level is also of interest and can be easily performed on microarray data, where the log ratios are centered around zero, by simply choosing a cut off value C and selecting all genes with absolute mean expression (or a certain proportion of samples with absolute expression) greater than C . With gene chip data such as this, the expression measurements span a large range of positive values so that choosing a cut off value for differential expression is harder. Hence, we used a variance based prescreen.

We proceeded to perform simultaneous clustering on the data set containing 2000 genes, ignoring the ALL/AML labels except to check how well the patient clusters corresponded with this classification.

6.1 *Clustering Genes within Patient Clusters*

First, we clustered patients using PAM with a Euclidean distance metric. The average silhouette is an output of the PAM algorithm which measures cluster strength. Maximizing the average silhouette is, therefore, one method for selecting the best number of clusters, k , from a range of options. This approach strongly suggested $k = 2$, and the resulting cluster labels corresponded very well with the ALL/AML labels (See Table 4). Next, we clustered genes within each of the two patient clusters. For both patient clusters, average silhouettes indicated two gene clusters and were very high for $k = 2$ (> 0.9). Figure 1 illustrates the results of this simultaneous clustering. The lines marking the cluster boundaries look sensible. We have reordered the patient and gene clusters and then the elements within each cluster according to the distance metric, so that those with similar expression profiles are close to each other. Note that in all plots the numbering of the genes on the axes refers to the reordering of genes for that patient cluster, so that a gene may appear in a different place in the ordering (i.e.: with a different number) for each patient cluster plot.

Within each patient cluster, we found one large cluster containing most genes and one small cluster containing the very highly expressed genes (plots 1a and 2a in Figure 1). There were 46 genes which made it into in the second cluster within both patient groups. In addition, there were 8 genes which were placed in gene cluster 2 within one patient cluster but *not* the other (6 for patient cluster 1 and 2 for patient cluster 2). While we would have found most of the 46 common genes and all of the 6 genes unique to patient cluster 1 if we had simply clustered genes using all patients, the 2 genes unique to patient cluster 2 would have been missed. Furthermore, we would likely not have realized that the 6 genes unique to patient cluster 1 were more highly expressed in these patients (mostly ALL) than the other patient group. This result is not surprising, since patient cluster 1 is more than twice the size of patient cluster 2, and hence its expression pattern could be expected to dominate overall. Here we have an illustration of the advantage of first clustering patients before clustering genes.

6.2 *Clustering Genes Hierarchically*

Another application of simultaneous clustering is to iteratively apply a gene (or patient) clustering algorithm in a hierarchical manner, so that at each

level the members of each cluster are themselves clustered. In this way, we can obtain smaller, more homogeneous clusters than those in the initial split. We have implemented this approach with PAM, using the average silhouette to select the number of clusters in each split. We can order the clusters at a given level according to the distance between their medoids. By carrying out the algorithm until every gene is in its own cluster or by stopping at a specified size cluster and then ordering the elements in each cluster, we obtain a unique final ordering of the elements. With genes, for example, the first split often shows strong evidence in favor of no more than two clusters, usually corresponding to over- and under-expressed genes. When these two subsets are examined separately, however, it is easy to see that the genes in each can be clustered into smaller groups. In some applications it is also interesting to hierarchically cluster samples. Even when one chooses to work with the first level of the tree only, the rundown tree can still provide a useful ordering of the elements.

We applied hierarchical PAM with a Euclidean distance metric to the ALL/AML data set. After the initial split of the genes into two clusters (corresponding to low and high average expression), we again clustered the genes in each of these gene clusters in a hierarchical fashion in order to obtain a final ordering of the genes. Figure 2 shows the ordered distance matrix. The blocks on the diagonal are clusters of similar genes. We can indicate the cluster splits from any level of the tree with lines. There were four clusters at level two of the tree and ten clusters at level three of the tree (one containing just one gene). The first split is most obvious, but it is easy to see why PAM divided the genes as it did at both of the next two levels. We find this sort of plot useful for deciding how many clusters there are in the data so that we can “cut” the tree at an appropriate level and possibly recombine clusters that visually do not look like they should have been split. This level can then be used to report gene clusters, possibly cluster patients within gene clusters, and bootstrap the result. Furthermore, if the clusters are strong, then they should correspond with very bright blocks on the diagonal. In this way, we can identify the most homogeneous clusters, which should contain genes that are part of the same causal pathway.

6.3 Clustering Patients within Gene Clusters

Finally, we performed the composition of gene and patient clustering in the opposite order. Using the gene clusters at any level of the hierarchical tree, patients can be clustered within gene clusters. The lower the level of the gene clustering tree, the more gene clusters there are to work with and hence the genes in each of these are fewer in number and more homogeneous. Although most clusters of genes showed little variance across patients, we found strong

evidence of patient clustering in some of the gene clusters.

We had imagined that the ALL/AML distinction might be explained by a single cluster of genes very differentially expressed between the two types. While gene clusters which partition patients well exist in this data set, no single gene cluster was able to cluster the ALL and AML patients as well as the full set of genes. In fact, even when we used our knowledge of the ALL/AML labels and selected the genes with largest t-statistics for the difference in means between the two groups, the top 65 genes were needed to cluster patients as well as the full set of genes. These genes came from several different gene clusters and had both low and high mean expression. Perhaps a different distance metric might have clustered these genes closer together, although we found that they fell into different gene clusters with both Euclidean and correlation metrics.

Although the patient clusters found in some of the small gene clusters do not correspond directly with the ALL/AML distinction, they are nonetheless interesting findings. It seems likely to us that they might correspond with some other clinical or pathological variables, such as tumor grade, survival, or treatment history. Using the ten gene clusters from level three, we found several small groups of genes with very strong evidence of patient clustering. In five of these gene clusters, we found groups of between one and four patients which clustered separately from all the others. In each case, the unique patients were either all ALL *or* all AML patients, leading us to believe that the ALL/AML categories themselves might be quite heterogeneous with subgroups of patients within each sharing particular mutations. In several cases, there was just one patient with a very different profile than the others across several genes (See Figure 3). Expression patterns such as these could be experimental errors (e.g.: bad slide regions) or in fact evidence of a different biological state, such as having a particular biochemical pathway activated. It would certainly be interesting to look for correlations between the gene cluster specific patient labels and various outcome variables.

By clustering patients within gene clusters we have found more complicated patterns than those seen when patients are simply clustered overall. This data set is known for the easy distinction of ALL and AML patients, and the finer level of patient clustering illustrated here is likely to correspond to other biological distinctions. We want to point out, however, that in other data sets we have worked with, the identification of groups of patients has been more complex in the sense that strong clusters were not evident overall (as they were here), but there were in fact smaller gene clusters which clustered patients well. Regardless of whether or not there is strong clustering of samples overall, the identification of small groups of genes with interesting expression patterns across samples can help researchers develop hypotheses to test in the lab or field.

6.4 Bootstrap Results

As we recommend, findings such as those discussed above should be qualified by reliability measures. In order to illustrate the way in which the bootstrap can be applied to perform inference on summary measures of a simultaneous clustering parameter, we looked at the result from clustering genes within patient clusters. We generated new samples using (i) nonparametric bootstrap, (ii) convex bootstrap with $d = 0.1$, and (iii) parametric bootstrap based on a mixture of two normal distributions with diagonal covariances. The empirical values of the mixing proportion, means, and variances from the two patient clusters in the original data were used in the parametric bootstrap.

For each bootstrap sample, the simultaneous clustering procedure was repeated: patients were clustered first, followed by genes within each patient cluster. We performed unsupervised clustering but used the same values for k as in the data analysis ($k = 2$ in each case). Since the two clusters from each application of PAM differed significantly in size in this case, we were able to infer a correspondence between the unsupervised clusters and the original clusters. We kept track of the medoids associated with the “big” and “small” clusters. These were very uniform in expression profile across bootstrap samples, particularly for gene clustering where the “small” cluster medoid always had much higher expression. One could also keep track for each gene of the proportion of bootstrap samples in which it appeared in each gene cluster. As discussed in van der Laan and Bryan [18], this measure estimates the probability that a gene belongs to its cluster. We also kept track of average silhouette and proportion of elements in the larger cluster.

Table 5 contains the results of the bootstrap analysis. The confidence intervals for the summary measures indicate that the clustering results, particularly for genes, were quite stable. We see some bias in the confidence intervals for summary measures of patient clustering (i.e.: they are not centered around the point estimate from the observed data), presumably due to the small sample size. Here, empirical confidence intervals might be better than normal distribution confidence intervals, since the central limit theorem does not appear to be in action yet.

By simulating data from an appropriate null distribution, it is possible to derive quantiles for summary measures under the null hypothesis that there is no clustering. These can be used to test whether summary measures show true evidence of clustering. We did this simulation for each clustering result and found the 0.95 quantiles of the average silhouette from single multivariate normals – not mixtures – with diagonal covariances and the empirical mean and variance vectors. For patient clustering, the lower 95% confidence intervals for the observed average silhouettes are only slightly larger than the null value,

indicating that the clustering of patients is barely significant with this sample size. For gene clustering within each patient cluster, however, the lower 95% confidence intervals for the observed average silhouettes were well above the null values, confirming that there is strong evidence of gene clustering in this data set.

7 Discussion

We have demonstrated that a large family of two-way clustering methods can be regarded as compositions of mappings involving clustering of genes and/or mappings involving clustering of patients. In this way, we can define a simultaneous clustering parameter as a function of the underlying data generating distribution that produced the results of a gene expression experiment. This statistical formalism allows us to understand and perform both estimation and inference using many commonly employed clustering methods. By forming iterative compositions, we can also design more aggressive algorithms for finding patterns in data. The beauty of this framework is that statistical inference, using methods such as the bootstrap, allows us to assess the reliability of patterns found by such “greedy” algorithms. In the context of gene expression data, where the dimension of the problem (p = number of genes) far exceeds the sample size (n), the need for this statistical rigor is particularly important and all too often overlooked by researchers keen to identify biological relationships between genes clustered together.

The two-way clustering methods discussed in Tibshirani *et al.* [17] (e.g.: hierarchical clustering, tree structured k-means, block clustering) are examples of commonly employed techniques which can be formally defined as simultaneous clustering parameters. Their gene shaving methodology is a more aggressive method with a specific goal: identification of small, homogeneous subsets of genes which have maximal variance across patients. We agree strongly with the rationale behind gene shaving, that different clusters of genes may cluster samples in different ways. This idea motivated us to employ PAM in a hierarchical fashion to first cluster genes and then again to cluster patients within gene clusters. One strength of hierarchical PAM over gene shaving is that it will identify clusters of genes with both low and high variance across patients. The lower variance clusters can be equally interesting biologically and (in some cases) can in fact produce interesting patient clustering results.

We have illustrated how bootstrap methods can be used to estimate the variability of simultaneous clustering parameters. Our simulations identified several interesting points about the bootstrap. Contrary to our initial hypothesis that the parametric bootstrap would be best able to approximate the distribution of an observed simultaneous clustering parameter θ_n , we found that both

the nonparametric and the parametric bootstraps performed well as methods for estimating variance in the context of gene and patient clustering. We were pleased to see that for sample sizes in the range of $n = 150$, the bootstrap is a valid method for estimate the variability of simultaneous clustering parameters. For a sample size as small as $n = 40$, however, the nonparametric bootstrap was conservative for estimating the variance of some summary measures of θ_n (particularly in unsupervised patient clustering). We were surprised to find that using convex pseudo-data was not an improvement over the nonparametric bootstrap. In light of these results and the computational ease of nonparametric resampling, we recommend the nonparametric bootstrap for inference with reasonable sample sizes. With sample sizes as small as $n = 40$, the researcher should be aware that the nonparametric bootstrap will be more conservative than the parametric.

The bootstrap is not only a method to estimate the distribution of an estimate, but it can also be viewed as a simulation study investigating the random behavior of an empirical estimate of θ under some known data generating distribution. In the statistics and biostatistics literature, simulation studies from standard parametric families are generally accepted as a way of obtaining insight into new estimation procedures. In particular, the bootstrap assesses how hard it is to estimate clustering parameters under a known law. Therefore, the output of the parametric bootstrap will provide valuable information even when the bootstrap distribution is not close to the true data generating distribution. A heavily discrete distribution such as P_n (nonparametric bootstrap) resulting in many ties may not be as insightful as a simulation from a smooth distribution such as a mixture of multivariate normal distributions. Furthermore, a parametric bootstrap allows for simulations from a null distribution so that testing can be performed, as illustrated for average silhouettes in the data analysis.

Although our aim was to examine the bootstraps as methods for estimating variance, we noted in passing that there was bias in point estimates from bootstrap samples, particularly for summary measures which were less smooth functions of the data generating distribution (e.g.: patient cluster means). We also found that for some parameters, normal approximation confidence intervals were inappropriate so that empirical quantiles should be used instead.

The analysis of gene expression data from Golub *et al.* [9] demonstrates that simultaneous clustering parameters identify complex patterns in gene expression data, including genes specifically up-regulated in subtypes of acute leukemia and clusters of genes which produce interesting patient clustering results not seen when all genes are used. Experience with other data sets supports the idea that simultaneous clustering provides additional insight over one-way clustering results. We have seen that even when the goal is to cluster samples, first clustering genes and then samples within gene clusters can help to

identify important relationships amongst samples that correspond with distinct biological states and to highlight genes which are members of a single biochemical/causal pathway. Pictures of the data matrix and particularly the distance matrix emphasize that these insights are of real interest, since simultaneous clustering labels correspond well with visual patterns seen in the data. Applying the bootstrap helps to understand the reliability of these patterns.

8 Acknowledgments

Both researchers were funded by Life Sciences Informatics Program Grant L98-10050 with industrial sponsor Chiron Corporation, Emeryville, CA.

References

- [1] Alizadeh A.A., Eisen M.B., Davis R.E., Ma C., Lossos I.S., Rosenwald A., Boldrick J.C., Sabet H., Tran T., Yu X., Powell J.I., Yang L., Marti G.E., Moore T., Hudson T. Jr., Lu L., Lewis D.B., Tibshirani R., Sherlock G., Chan W.C., Greiner T.C., Weisenberger D.D., Armitage J.O., Warnke R., Levy R., Wilson W., Grever M.R., Byrd J.C., Botstein D., Brown P.O., Staudt L.M., Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, *Nature* **403** (2000) pp. 503–511.
- [2] Bittner M., Meltzer P., Chen Y., Jiang Y., Seftor E., Hendrix M., Radmacher M., Simon R., Yakhini Z., Ben-Dor A., Sampas N., Dougherty E., Wang E., Marincola F., Gooden C., Lueders J., Glatfelter A., Pollock P., Carpten J., Gillanders E., Leja D., Dietrich K., Beaudry C., Berens M., Alberts D., Sondak V., Hayward N., Trent J., Molecular classification of cutaneous malignant melanoma by gene expression profiling, *Nature* **406** (2000) pp. 536–540.
- [3] Breiman L., Using convex pseudo-data to increase prediction accuracy, Technical Report no. 513, U. C. Berkeley, Dept. of Statistics, March 1998.
- [4] Debouck C. and Goodfellow P.N., DNA microarrays in drug discovery and development, *Nature Genetics* **21**: 1, suppl. (1999) pp. 48–50.
- [5] DeRisi J., Penland L., Brown P.O., Bittner M.L., Meltzer P.S., Ray M., Chen Y., Su Y.A., Trent J.M., Use of a cDNA microarray to analyse gene expression patterns in human cancer, *Nature Genetics* **14** (1996) pp. 456–460.
- [6] Eisen M.B., Spellman P.T., Brown P.O., Botstein D., Cluster analysis and display of genome-wide expression patterns, *PNAS* **95** (1998) pp. 14863–14868.
- [7] Fraley C. and Raftery A.E., Model-based clustering, discriminant analysis, and density estimation, Technical Report no. 380, Univ. of Washington, Dept. of Statistics, Oct. 2000.

- [8] Giné E. and Zinn J., Bootstrapping general empirical measures, *Ann. Probability* **18** (1990) pp. 851–869.
- [9] 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., Lander E.S., Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring, *Science* **286** (1999) pp. 321–531.
- [10] Hughes T.R., Marton M.J., Jones A.R., Roberts C.J., Stoughton R., Armour C.D., Bennett H.A., Coffey E., Dai H., He Y.D., Kidd M.J., King A.M., Meyer M.R., Slade D., Lum P.Y., Stepaniants S.B., Shoemaker D.D., Gachotte D., Chakraborty K., Simon J., Bard M., Friend S.H., Functional discovery via a compendium of expression profiles, *Cell* **102** (2000) pp. 109–126.
- [11] Kaufman L. and Rousseeuw P.J., *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley & Sons (New York, 1990).
- [12] Lillie J., Probing the genome for new drugs and targets with DNA arrays, *Drug Development Research* **41** (1997) pp. 160–172.
- [13] Lockhart D.J. and Winzler E.A., Genomics, gene expression and DNA arrays, *Nature* **405** (2000) pp. 827–836.
- [14] Marton M.J., DeRisi J.L., Bennett H.A., Iyer V.R., Meyer M.R., Roberts C.J., Stoughton R., Burchard J., Slade D., Dai H., Bassett D.E. Jr., Hartwell L.H., Brown P.O., Friend S.H., Drug target validation and identification of secondary drug target effects using DNA microarrays, *Nature Medicine* **4** (1998) pp. 1293–1301.
- [15] Perou C.M., Jeffrey S.S., Van de Rijn M., Rees C.A., Eisen M.B., Ross D.T., Pergamenschikov A., Williams C.F., Zhu S.X., Lee J.C.F., Lashkari O., Shalon D., Brown P.O., Botstein D., Distinctive gene expression patterns in human mammary epithelial cells and breast cancers, *Proc. Natl. Acad. Sci.* **96** (1999) pp. 9212–9217.
- [16] 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., Brown P.O., Systematic variation in gene expression patterns in human cancer cell lines, *Nature Genetics* **24** (2000) pp. 227–235.
- [17] Tibshirani R., Hastie T., Eisen M., Ross D., Botstein D., Brown P., Clustering methods for the analysis of DNA microarray data. Technical Report, Stanford University, Oct. 1999.
- [18] van der Laan M.J. and Bryan J.F. Gene Expression Analysis with the Parametric Bootstrap, Technical Report no. 81, U. C. Berkeley, Group in Biostatistics, Jan. 2000.

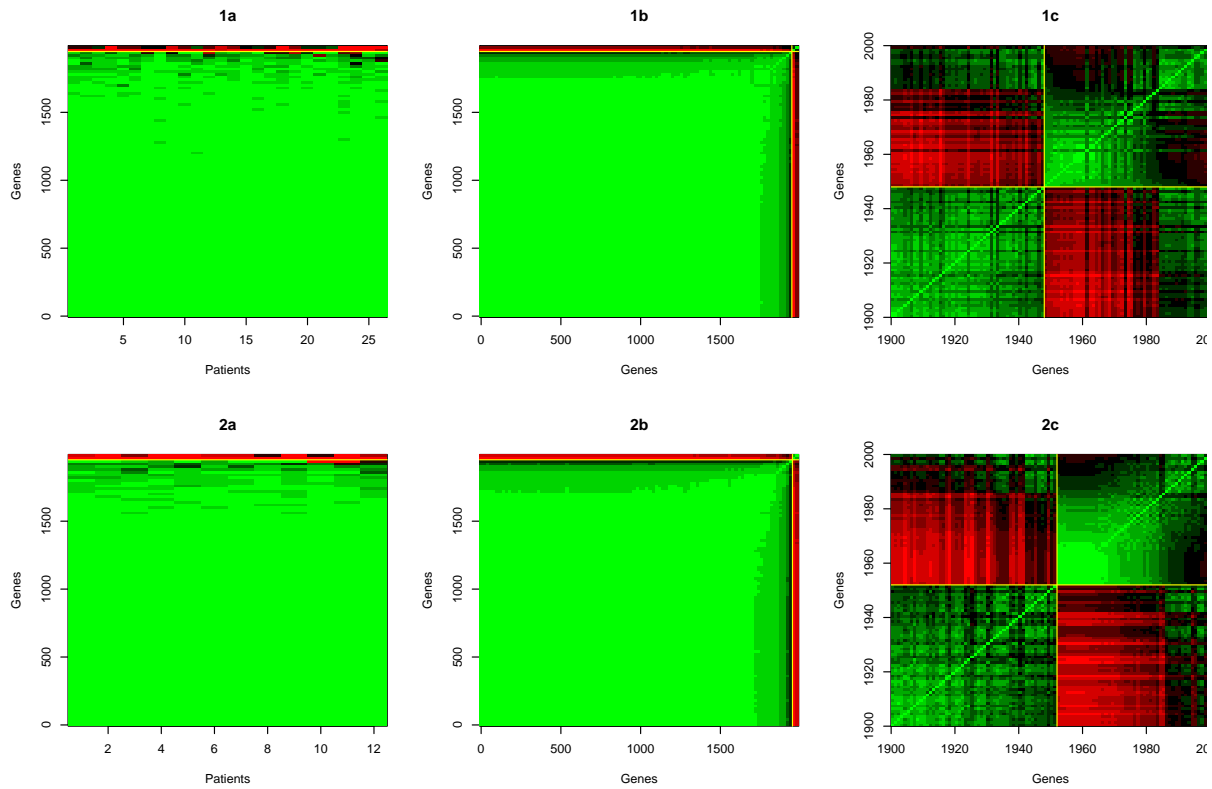


Fig. 1. Results of simultaneous clustering of patients and then genes within patient clusters 1 (1a-c) and 2 (2a-c). Lines mark the division between gene clusters. Plots 1a and 2a show the reordered data matrices. Each intensity is represented by a color on the green-red scale with bright green corresponding to the lowest expression and bright red to the highest. Plots 1b and 2b show the reordered distance matrices for all genes based on the patients in that patient cluster. Green corresponds to the lowest distance (i.e.: most similar genes) and red to the highest distance. Plots 1c and 2c show the upper right hand corners of the distance matrices in more detail.

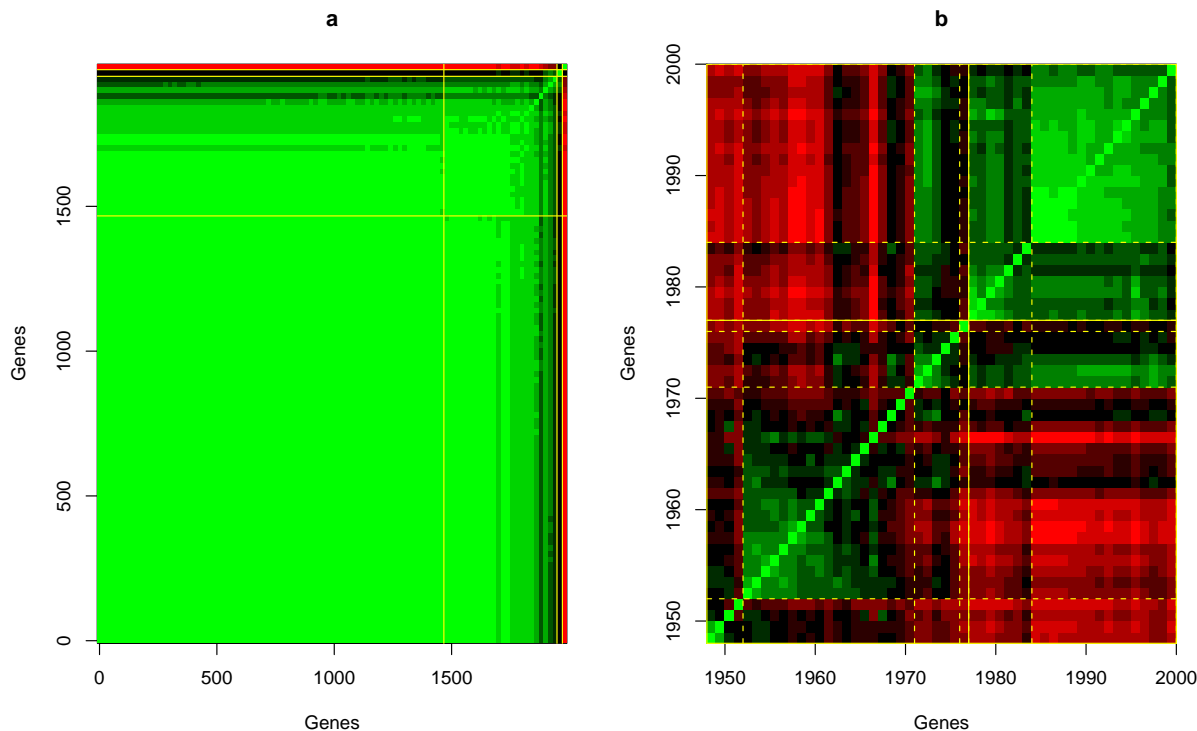


Fig. 2. Results of hierarchical PAM on genes. (a) The reordered distance matrix is plotted with the cluster boundaries from the second level of the tree (four gene clusters). (b) A more detailed view of the upper right hand corner of the distance matrix, containing all genes assigned to cluster 2 in the first level of the tree (the high expression group). The solid lines are the cluster boundaries from the second level of the tree (as in (a)). The dotted lines are the additional cluster boundaries added in the third level of the tree. The gene labeled 1977 is in a cluster by itself in the third level. The stronger the similarity between the genes in a cluster, the more solid that cluster will appear. The last cluster (genes labeled 1985-2000), for example, is a very solid block on the diagonal. The intensities of these genes are consistently very high across patients and could be interesting drug targets. In both plots, green corresponds to the lowest distance (i.e.: similar genes) and red to the highest.

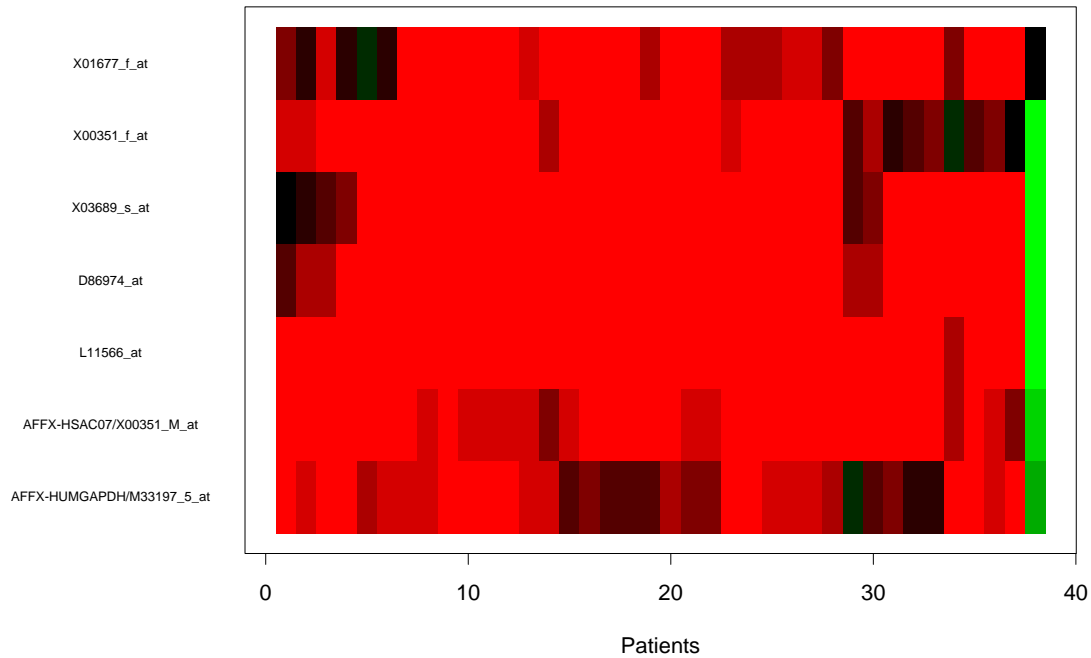


Fig. 3. Example of the data matrix for a gene cluster with strong evidence of patient clustering. Each intensity is represented by a color on the green-red scale with bright green corresponding to the lowest expression and bright red to the highest. The patients have been reordered using only the genes in this cluster. In this case, one patient has a very different expression pattern from the others. The fact that this patient is not expressing these genes at the high levels seen in all the other patients could be correlated with some clinical or pathological variable. It is also interesting to note how similar the genes are to each other. This cluster is the first from the left in Figure 2b.

	0.9 quantile of maximum absolute difference		
Parameter:	Mean	Median	Correlation
n=40			
True distribution	0.60	0.74	0.75
Nonparametric	0.60	0.98	0.89
Convex d=0.1	0.59	0.97	0.89
Convex d=0.3	0.54	0.86	0.88
Convex d=0.5	0.50	0.78	0.87
Parametric	0.63	0.93	0.84
n=250			
True distribution	0.25	0.30	0.35
Nonparametric	0.26	0.38	0.36
Convex d=0.1	0.23	0.34	0.36
Convex d=0.3	0.21	0.30	0.36
Convex d=0.5	0.20	0.27	0.36
Parametric	0.24	0.36	0.35

Table 1

Results of Simulation 1 for gene clustering. $B = 100$ i.i.d. bootstrap samples were used in each simulation. Every bootstrap sample included $n = 40$ or $n = 250$ observations of a 3000-dimensional gene expression vector. The 0.9 quantile of the maximum absolute difference in each summary measure is reported.

	0.9 quantile of maximum absolute difference		0.95 CI, $\sqrt{n}\hat{\sigma}$
Parameter:	Mean 1	Mean 2	\hat{q}
n=40, p=3000			
True distribution	1.31	0.79	{0.16,0.44}, 0.45
Nonparametric	1.21	0.87	{0.22,0.49}, 0.44
Convex d=0.1	1.19	0.83	{0.21,0.48}, 0.43
Convex d=0.3	1.05	0.79	{0.22,0.50}, 0.45
Convex d=0.5	0.99	0.68	{0.20,0.49}, 0.46
Parametric	1.41	0.94	{0.20,0.49}, 0.46
n=250, p=3000			
True distribution	0.50	0.32	{0.25,0.36}, 0.45
Nonparametric	0.51	0.32	{0.24,0.34}, 0.41
Convex d=0.1	0.47	0.29	{0.24,0.35}, 0.46
Convex d=0.3	0.43	0.26	{0.25,0.35}, 0.40
Convex d=0.5	0.42	0.26	{0.24,0.36}, 0.47
Parametric	0.52	0.33	{0.25,0.33}, 0.43
n=150, p=12			
True distribution	0.38	0.25	{0.25,0.41}, 0.49
Nonparametric	0.51	0.26	{0.25,0.40}, 0.46
Convex d=0.1	0.36	0.27	{0.24,0.39}, 0.46
Convex d=0.3	0.33	0.24	{0.23,0.37}, 0.44
Convex d=0.5	0.32	0.20	{0.23,0.36}, 0.41
Parametric	0.37	0.28	{0.28,0.44}, 0.49

Table 2

Results of Simulation 2 for supervised clustering of patients. $B = 100$ i.i.d. bootstrap samples were used in each simulation. Every bootstrap sample included $n = 40$ or $n = 250$ observations of a 3000-dimensional gene expression vector or $n = 150$ observations of a 12-dimensional gene expression vector. The 0.9 quantile of the maximum absolute difference in each cluster's mean vector is reported. 95% confidence intervals for the proportion of patients in the first cluster were calculated using a normal approximation. We also report \sqrt{n} times the estimated standard error of \hat{q} from each bootstrap sample.

	0.95 CI, $\sqrt{n}\hat{\sigma}$		
Parameter:	Dist. Between Medoids	Diam. Smaller Clust.	Avg. Silhouette
n=40, p=3000			
True dist.	{85.70,89.12}, 5.51	{78.64,80.55}, 3.07	{0.12,0.13}, 0.012
Nonparametric	{83.22,92.36}, 14.75	{75.67,83.82}, 13.15	{0.12,0.21}, 0.13
Convex d=0.1	{75.52,86.86}, 18.30	{50.61,100.11,83.82}, 79.86	{0.12,0.20}, 0.14
Convex d=0.3	{61.62,79.04}, 28.11	{33.02,106.67}, 118.82	{0.076,0.19}, 0.19
Convex d=0.5	{53.62,74.26}, 33.29	{41.72,100.03}, 94.06	{0.049,0.17}, 0.19
Parametric	{86.96,91.06}, 6.62	{78.76,80.95}, 3.50	{0.14,0.15}, 0.012
n=250, p=3000			
True dist.	{84.79,88.76}, 16.02	{80.37,81.43}, 4.31	{0.126,0.128}, 0.010
Nonparametric	{85.85,89.91}, 16.34	{79.91,80.77}, 3.47	{0.13,0.14}, 0.017
Convex d=0.1	{76.84,83.70}, 27.69	{78.06,79.79}, 6.96	{0.13,0.14}, 0.023
Convex d=0.3	{62.49,77.71}, 61.36	{76.32,80.48}, 16.76	{0.13,0.14}, 0.080
Convex d=0.5	{54.26,74.00}, 79.60	{75.87,79.89}, 16.22	{0.075,0.13}, 0.20
Parametric	{84.69,88.88}, 16.90	{80.27,81.49}, 4.92	{0.130,0.132}, 0.0084
n=150, p=12			
True dist.	{3.57,3.77}, 5.94	{7.55,9.32}, 5.54	{0.033,0.15}, 0.35
Nonparametric	{3.05,5.06}, 6.29	{7.35,9.09}, 5.42	{0.066,0.14}, 0.22
Convex d=0.1	{3.03,4.70}, 5.22	{6.97,8.79}, 5.67	{0.064,0.14}, 0.22
Convex d=0.3	{2.34,4.05}, 5.33	{6.47,8.44}, 6.16	{0.048,0.13}, 0.25
Convex d=0.5	{2.04,3.78}, 5.42	{6.19,7.93}, 5.44	{0.047,0.13}, 0.26
Parametric	{2.67,4.66}, 6.23	{7.70,9.27}, 4.89	{0.045,0.13}, 0.26

Table 3

Results of Simulation 2 for unsupervised clustering of patients. $B = 100$ i.i.d. bootstrap samples were used in each simulation. Every bootstrap sample included $n = 40$ or $n = 250$ observations of a 3000-dimensional gene expression vector. The normal approximation 95% confidence interval for each summary measure is reported. We also report \sqrt{n} times the estimated standard error of the summary measures from each bootstrap sample. Three significant figures are reported where needed to see the width of very narrow confidence intervals.

	ALL	AML	
Cluster 1	25	1	26
Cluster 2	2	10	12
	27	11	38

Table 4

Clustering of patients using PAM with a Euclidean distance metric. The correspondence between the clustering labels and the ALL/AML labels is strong.

	0.95 CI, $\sqrt{n}\hat{\sigma}$		
	Patients	Genes 1	Genes 2
Prop. in larger cluster \hat{q} :			
Observed Data	0.6842	0.9740	0.9760
Nonparametric	{0.53,0.99},0.72	{0.327,0.332},0.0084	{0.32,0.33},0.018
Convex d=0.1	{0.57,0.95},0.60	{0.32,0.33},0.0077	{0.32,0.33},0.013
Parametric	{0.0.58,0.89},0.48	{0.327,0.331},0.0077	{0.32,0.34},0.022
Silhouettes:			
Null 0.95 quantile	0.0224	0.0355	0.0851
Observed Data	0.1037	0.9189	0.9213
Nonparametric	{0.061,0.36},0.47	{0.91,0.94},0.050	{0.90,0.95},0.078
Convex d=0.1	{0.046,0.33},0.45	{0.91,0.94},0.051	{0.91,0.94},0.060
Parametric	{0.11,0.21},0.16	{0.91,0.93},0.024	{0.89,0.95},0.11

Table 5

Results of unsupervised bootstrap on ALL/AML data set ($n = 38$, $p = 2000$). $B = 100$ bootstrap samples were used to estimate normal approximation 95% confidence intervals for each summary measure. We also report \sqrt{n} times the estimated standard error of the summary measures from each bootstrap sample. Genes 1 and Genes 2 refer to results from clustering genes within patient clusters 1 and 2, respectively. The null 0.95 quantiles of the average silhouettes were computed from simulations with $B = 100$ samples drawn from null distributions. Three significant figures are reported where needed to see the width of very narrow confidence intervals.