An empirical study on Principal Component Analysis for clustering gene expression data

Ka Yee Yeung
Walter L. Ruzzo

Technical Report UW-CSE-2000-11-03
November, 2000

Department of Computer Science & Engineering
University of Washington
Seattle, WA 98195

An empirical study on Principal Component Analysis for clustering gene expression data

Ka Yee Yeung, Walter L. Ruzzo
Dept of Computer Science and Engineering, University of Washington
{kayee, ruzzo}@cs.washington.edu

Nov 1, 2000

Abstract

There is a great need to develop analytical methodology to analyze and to exploit the information contained in gene expression data. Because of the large number of genes and the complexity of biological networks, clustering is a useful exploratory technique for analysis of gene expression data. Other classical techniques, such as principal component analysis (PCA), have also been applied to analyze gene expression data. Using different data analysis techniques and different clustering algorithms to analyze the same data set can lead to very different conclusions. Our goal is to study the effectiveness of principal components (PC's) in capturing cluster structure. In other words, we empirically compared the quality of clusters obtained from the original data set to the quality of clusters obtained from clustering the PC's using both real gene expression data sets and synthetic data sets.

Our empirical study showed that clustering with the PC's instead of the original variables does not necessarily improve cluster quality. In particular, the first few PC's (which contain most of the variation in the data) do not necessarily capture most of the cluster structure. We also showed that clustering with PC's has different impact on different algorithms and different similarity metrics.

1 Introduction

DNA microarrays offer the first great hope to study variations of many genes simultaneously [Lander, 1999]. Large amounts of gene expression data have been generated by researchers. There is a great need to develop analytical methodology to analyze and to exploit the information contained in gene expression data [Lander, 1999]. Clustering analysis attempts to divide objects into groups such that objects within the same group are more similar to each other than objects in other groups. Because of the large number of genes and the complexity of biological networks, clustering is a useful exploratory technique for analysis of gene expression data. Since genes with related functions tend to have similar expression patterns, possible roles for genes with unknown functions can be suggested based on the known functions of some other genes that are placed in the same cluster [Chu *et al.*, 1998]. Clustering is sometimes used as a preprocessing step in inferring regulatory networks. For example, [Chen *et al.*, 1999] used clustering to identify ORF's that have similar expression patterns to reduce the size of the regulatory network to be inferred.

Many clustering algorithms have been proposed for gene expression data. For example, [Eisen *et al.*, 1998] applied the average-link hierarchical clustering algorithm to identify groups of co-regulated yeast genes. [Ben-Dor and Yakhini, 1999] reported success with their CAST algorithm. [Tamayo *et al.*, 1999] used self-organizing maps to identify clusters in the yeast cell cycle and human hematopoietic differentiation data sets. Other techniques, such as principal component analysis (PCA), have also been proposed to analyze gene expression data. Principal component analysis (PCA) ([Dunteman, 1989], [Everitt and Dunn, 1992], [Jolliffe, 1986]) is a classical technique to reduce the dimensionality of the data set by transforming to a new set of variables to summarize the features of the data set. In particular, [Raychaudhuri *et al.*, 2000] applied PCA to the sporulation data set [1].

Using different data analysis techniques and different clustering algorithms to analyze the same data set can lead to very different conclusions. For example, [Chu *et al.*, 1998] identified seven clusters in the sporulation data set using the Cluster software [Eisen *et al.*, 1998], but [Raychaudhuri *et al.*, 2000] claimed that there are no clusters present in the same data set when the data points are viewed in the space of the first two principal components (PC's). In this paper, we empirically investigate the effectiveness of PCA as a preprocessing step in cluster analysis using both real gene expression data sets with external clustering criteria and synthetic data sets.

2 Principal Component Analysis (PCA)

2.1 An Example of PCA

The central idea of principal component analysis (PCA) is to reduce the dimensionality of the data set while retaining as much as possible the variation in the data set. Principal components (PC's) are linear transformations of the original set of variables. PC's are uncorrelated and ordered so that the first few PC's contain most of the variations in the original data set [Jolliffe, 1986].

The first PC has the geometric interpretation that it is a new coordinate axis that maximizes the variation of the projections of the data points on the new coordinate axis. Figure 1 shows a scatterplot of some fictitious data points in two dimensions (x_1 and x_2). The points show an elliptical shape, and the first PC is in the direction of the principal axis of this ellipse (marked PC_1 in Figure 1). The second

[1] Sporulation is the process in which diploid cells undergo meiosis to produce haploid cells in reproduction of yeast. The sporulation data set [Chu *et al.*, 1998] shows the temporal expression patterns of 97% of yeast genes over seven successive time points in the sporulation of yeast.

PC is orthogonal to the first PC and is marked PC_2 in Figure 1. If the data points are projected onto the first PC, most of the variation of the two dimensional data points would be captured in one dimension.

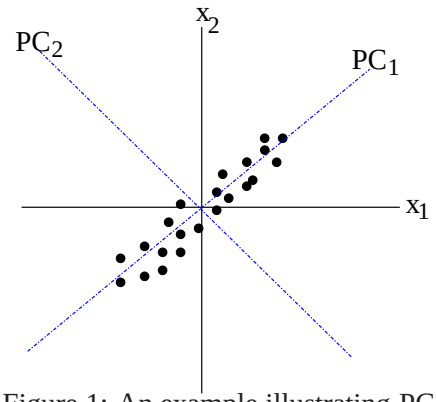

Figure 1: An example illustrating PCA

2.2 Definitions of PCA

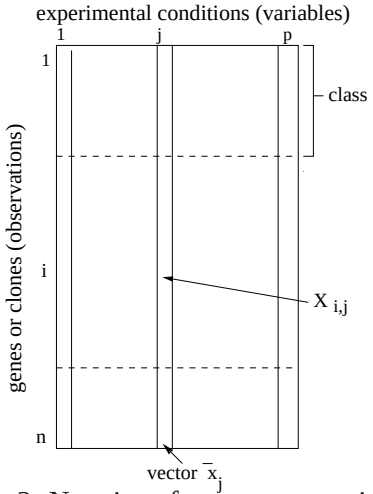

Figure 2: Notations for gene expression data

Let X be a gene expression data set with n genes and p experimental conditions. In this report, our goal is to cluster the genes in the data set, hence the experimental conditions are the variables. Let $\bar{\mathbf{x}}_\mathbf{j}$ be a column vector of expression levels of all the n genes under experimental condition j (see Figure 2). A PC is a linear transformation of the experimental conditions. Let $\bar{\mathbf{z}}_\mathbf{k} = \sum_{j=1}^{p} \alpha_{k,j} \bar{\mathbf{x}}_\mathbf{j}$ be the kth PC. In particular, the first PC, $\bar{\mathbf{z}}_\mathbf{1}$, can be written as $\sum_{j=1}^{p} \alpha_{1,j} \bar{\mathbf{x}}_\mathbf{j}$. Let Σ be the covariance matrix of the data, and $\bar{\alpha}_\mathbf{k}$ be a column vector of all the $\alpha_{k,j}$'s, *i.e.*, $\bar{\alpha}_\mathbf{k}^\mathbf{T} = (\alpha_{k,1}, \alpha_{k,2}, \ldots, \alpha_{k,p})$. To derive the first PC, we have to find $\bar{\alpha}_\mathbf{1}$ that maximizes $var(\sum_{j=1}^{p} \alpha_{1,j} \bar{\mathbf{x}}_\mathbf{j}) = \bar{\alpha}_\mathbf{1}^\mathbf{T} \Sigma \bar{\alpha}_\mathbf{1}$, subject to the constraint $\bar{\alpha}_\mathbf{1}^\mathbf{T} \bar{\alpha}_\mathbf{1} = 1$. It can be shown that $\bar{\alpha}_\mathbf{1}$ is the eigenvector corresponding to the largest eigenvalue, λ_1, of Σ, and $var(\bar{\mathbf{z}}_\mathbf{1}) = \lambda_1$ [Jolliffe, 1986]. In general, the kth PC, $\bar{\mathbf{z}}_\mathbf{k} = \sum_{j=1}^{p} \alpha_{k,j} \bar{\mathbf{x}}_\mathbf{j}$, can be derived by maximizing $var(\sum_{j=1}^{p} \alpha_{k,j} \bar{\mathbf{x}}_\mathbf{j})$, such that $\bar{\alpha}_\mathbf{k}^\mathbf{T} \bar{\alpha}_\mathbf{k} = 1$ and $\bar{\alpha}_\mathbf{k}^\mathbf{T} \bar{\alpha}_\mathbf{i} = 0$, where $i < k$. It can be shown that $\bar{\alpha}_\mathbf{k}$ is an eigenvector of Σ corresponding to its kth largest eigenvalue λ_k, and $var(\bar{\mathbf{z}}_\mathbf{k}) = \lambda_k$ [Jolliffe, 1986].

In the case of gene expression data, the population covariance matrix Σ is not known. The sample covariance matrix S can be used instead. Let $x_{i,j}$ be the gene expression level of gene i under experimental condition j. The sample covariance between conditions j and k, $S(j,k)$, can be calculated as $\frac{1}{n-1}\sum_{i=1}^{n}(x_{i,j}-\mu_{x_j})(x_{i,k}-\mu_{x_k})$, where $\mu_{x_j}=\frac{1}{n}\sum_{i=1}^{n}x_{i,j}$.

From the derivation of PC's, the kth PC can be interpreted as the direction that maximizes the variation of the projections of the data points such that it is orthogonal to the first $k-1$ PC's, and the kth PC has the kth largest variance among all PC's. Since most of the variation of high dimensional data points can be captured in reduced dimensions defined by the first few PC's, PCA is often used in visualization of high dimensional data points.

2.3 Choosing the number of PC's

Since the variance of PC's are ordered, usually the first m ($m \leq p$, where p is the number of experiments in the data set) PC's are used in data analysis. The next question is how we should choose m, the number of first PC's to be retained, to adequately represent the data set. There are some common rules of thumb to choose the number of components to retain in PCA. Most of the rules are informal and ad-hoc. The first common rule of thumb is to choose m to be the smallest integer such that a chosen percentage of total variation is exceeded. In [Raychaudhuri *et al.*, 2000], the first two components which represent over 90% of the total variation in the sporulation data were chosen. Another common approach uses a *scree graph*, in which the kth eigenvalue is plotted against the component number, k. The number of components m is chosen to be the point at which the line in the scree graph is "steep" to the left but "not steep" to the right. The main problem with these approaches is that they are very subjective. There are some more formal approaches in the literature, but in practice, they tend not to work as well as the ad-hoc approach [Jolliffe, 1986].

2.4 Covariance versus correlation matrices

In the PCA literature, some authors prefer to define PC's using the *correlation* matrix instead of the covariance matrix. The correlation between a pair of variables is equivalent to the covariance divided by the product of the standard deviations of the two variables. Extracting the PC's as the eigenvectors of the correlation matrix is equivalent to computing the PC's from the original variables after each has been standardized to have unit variance. PCA based on covariance matrices has the potential drawback that the PC's are highly sensitive to the unit of measurement. If there are large differences between the variances of the variables, then the first few PC's computed with the covariance matrix are dominated by the variables with large variances. On the other hand, defining PC's with the correlation matrix has the drawback that the data is arbitrarily re-scaled to have unit variance. The general rule of thumb is to define PC's using the correlation matrix if the variables are of different types [Jolliffe, 1986].

2.5 Application of PCA in cluster analysis

In the clustering literature, PCA is sometimes applied to reduce the dimension of the data set prior to clustering. The first few (say m, $m \leq p$) PC's are usually used (for example, [Jolliffe *et al.*, 1980]). Most clustering algorithms require a measure of pairwise similarity or dissimilarity between observations as input. There are two popular pairwise similarity metrics in clustering gene expression data: Euclidean distance (for example, [Wen *et al.*, 1998]) and correlation coefficient (for example, [Eisen *et al.*, 1998]). The pairwise Euclidean distance between two objects is unchanged after the PCA step if all p PC's are used. When Euclidean distance is used as the similarity metric, using the first

m PC's simply provides an approximation to the similarity metric [Jolliffe, 1986]. When correlation coefficient is used as the similarity metric, the pairwise correlation coefficient between two objects is not the same after the PCA step even if all p PC's are used. There is no simple relationship between the correlation coefficients of the same pair of objects with and without PCA.

In general, the extra computation to find the PC's far outweighs any reduction in running time for using fewer PC's to compute the Euclidean distance [Jolliffe, 1986]. So, the hope for using PCA prior to cluster analysis is that PC's may "extract" the cluster structure in the data set. Figure 3 is a fictitious situation in which the PCA preprocessing step before cluster analysis may help. The first PC is in the direction of inter-cluster separation (the blue dotted line) in Figure 3. Projection of the data points on the first PC clearly highlights the separation between the two clusters in the data. However, PCA does not help in all situations. For example, in Figure 4, the first PC is in the direction of x_2. Projection of the data points onto the first PC does not preserve the separation between the two clusters in the data.

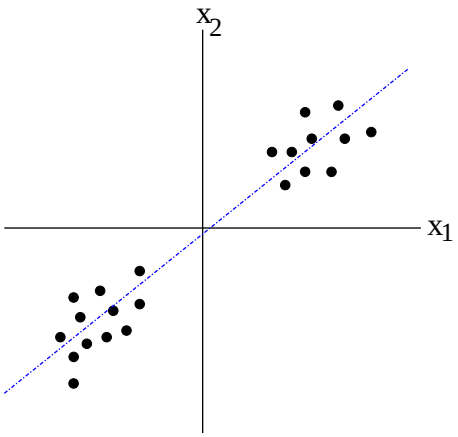

Figure 3: An example illustrating PCA helps in cluster analysis.

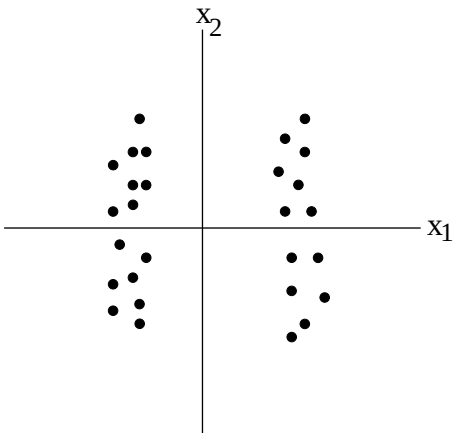

Figure 4: An example illustrating PCA does not help in cluster analysis.

In addition to the fictitious examples above illustrating the possible pros and cons of PCA on cluster analysis, [Chang, 1983] showed theoretically that the first few PC's may not contain cluster information under certain assumptions. Assuming that the data is a mixture of two multivariate normal distributions with different means but with an identical within-cluster covariance matrix, [Chang, 1983] derived a relationship between the distance of the two subpopulations and any subset of PC's, showing that the set of PC's with the largest eigenvalues does not necessarily contain more cluster structure information (the distance between the two subpopulations is used as a measure of discriminatory power for cluster structures). He also generated an artificial example in which there are two classes, and if the data points are visualized in two dimensions, the two classes are only well-separated in the subspace of the first and last PC's.

In [Raychaudhuri *et al.*, 2000], PCA was applied to the sporulation data set [Chu *et al.*, 1998]. The data points were visualized in the subspace of the first two PC's, and they showed a unimodal distribution. [Raychaudhuri *et al.*, 2000] concluded that the sporulation data may not contain any clusters based on visualization. With [Chang, 1983]'s theoretic results and the possibility of the situation in Figure 4 in mind, it is clear that clustering with the PC's instead of the original variables does not have universal success. However, the theoretical results in [Chang, 1983] are true only under an unrealistic assumption for gene expression data (*i.e.*, there are two classes and each of the classes is generated according to the multivariate normal distribution with a common covariance matrix). Therefore, there is a need to investigate the effectiveness of PCA as a preprocessing step to cluster analysis on gene

expression data before any conclusions are drawn. Our report is an attempt for such an empirical study.

3 Overview of Our Methodology

Our goal is to empirically investigate the effectiveness of clustering gene expression data using PC's instead of the original variables. Our methodology is to run a clustering algorithm on a given data set, and then apply the same algorithm to the PC's of the same data set. Then, the clustering results with and without PCA are compared against an external criterion. The details of the experiments will be discussed in the following sections.

3.1 Data sets

We used two gene expression data sets which have external criteria, and four sets of synthetic data to evaluate the effectiveness of PCA. In this report, we use the word *class* to refer to a group in our external criteria that is used to assess clustering results. The word *cluster* is used to refer to clusters obtained by a clustering algorithm. We assume both classes and clusters are disjoint.

3.1.1 Gene expression data sets

The ovary data: A subset of the ovary data set ([Schummer *et al.*, 1999], [Schummer, 2000]) is used. The ovary data set is generated by hybridizing randomly selected cDNA's to membrane arrays. The subset of the ovary data set we used contains 235 clones (clones are portions of genes) and 24 samples, 7 of which are derived from normal tissues, 4 from blood samples, and the remaining 13 from ovarian cancers in various stages of malignancy. The tissue samples are the experimental conditions. The 235 clones were sequenced, and they correspond to 4 different genes. The numbers of clones corresponding to each of the four genes are 58, 88, 57, and 32 respectively. We expect clustering algorithms to separate the four different genes. Hence, the four genes form the four classes of external criteria for this data set. Different clones may have different hybridization intensities. Therefore, the data for each clone is normalized across the 24 experiments to have mean 0 and variance 1 [2].

The yeast cell cycle data: The second gene expression data set we used is the yeast cell cycle data set [Cho *et al.*, 1998] which shows the fluctuation of expression levels of approximately 6000 genes over two cell cycles (17 time points). By visual inspection of the raw data, [Cho *et al.*, 1998] identified 420 genes which peak at different time points and categorized them into five phases of cell cycle. Out of the 420 genes they classified, 380 genes were classified into only one phase (some genes peak at more than one phase in the cell cycle). Since the 380 genes were identified by visual inspection of gene expression data according to the peak times of genes, we expect clustering results to correspond to the five phases to a certain degree. Hence, we used the 380 genes that belong to only one class (phase) as our external criteria. The 17 time points are the experimental conditions. The data is normalized to have mean 0 and variance 1 across each cell cycle as suggested in [Tamayo *et al.*, 1999].

[2] If the correlation matrix is used instead of the covariance matrix in PCA, the tissue samples (experiments) are the variables and are standardized across all the clones.

3.1.2 Synthetic data sets

Since the array technology is still in its infancy [Lander, 1999], the "real" data may be noisy, and clustering algorithms may not be able to extract all the classes contained in the data. There may also be information in real data that is not known to biologists. Furthermore, synthetic data sets provide us with inexpensive replicates of the data to increase the reliability of our empirical study. Therefore, we would like to complement our empirical study of the effectiveness of PCA with synthetic data, for which the classes are known.

To the best of our knowledge, modeling gene expression data sets is an ongoing effort by many researchers, and there is no well-established model to represent gene expression data. The following four sets of synthetic data represent our preliminary efforts on synthetic gene expression data generation. We do not claim that any of the four sets of synthetic data capture most or all of the characteristics of gene expression data. Each of the synthetic data has strengths and weaknesses. By using *all* four sets of synthetic data to evaluate the effectiveness of PCA on clustering, we hope to achieve a thorough comparison study capturing many different aspects of gene expression data.

The first two synthetic data sets represent attempts to generate replicates of the ovary data set by randomizing different aspects of the original data. The last two synthetic data sets are generated by modeling expression data with a mathematical formula. In each of the four synthetic data sets, ten replicates are generated. Ideally, more replicates would be more desirable. However, the algorithms have very long running time[3]. In each replicate, 235 observations and 24 variables are randomly generated.

Mixture of normal distributions on the ovary data: Visual inspection of the ovary data suggests that the data is not too far from normal. The expression levels for different clones of the same gene are not identical due to the fact that the clones represent different portions of the cDNA. Figure 5 shows the distribution of the expression levels in a normal tissue in a different class (gene) from the ovary data. We found that the distributions of the normal tissue samples are typically closer to normal distributions than those of tumor samples, for example, Figure 6. Even though some of the tumor tissues from some classes (genes) do not closely follow the normal distribution, we generate the data using a mixture of multivariate normal distributions in this synthetic data set.

The sample covariance matrix and the mean vector of each of the four classes (genes) in the ovary data are computed. The size of each class in the synthetic data is the same as the ovary data. Each class in the synthetic data is generated according to a multivariate normal distribution with the sample covariance matrix and the mean vector of the corresponding class in the ovary data.

This synthetic data set preserves the covariance between the tissue samples in each gene. It also preserves the mean vectors of each class. The weakness of this synthetic data set is that the assumption of the underlying multivariate normal distribution for each class may not be true for real genes.

Randomly permuted ovary data: No underlying distribution of the gene expression data is assumed in this synthetic data set. The size of each class in this synthetic data set is again the same as the ovary data. The random data for an artificial gene in class c (where $c = 1, \ldots, 4$) under experimental condition j (where $j = 1, \ldots, 24$) can be generated by randomly sampling (with replacement) the expression levels under condition j in the same class c of the ovary data.

This data set does not assume any underlying distribution. However, any possible correlation between tissue samples (for example, the normal tissue samples may be correlated) is not preserved due to the independent random sampling of the expression levels from each experimental condition.

[3] It takes approximately 3 to 4 hours to run the modified greedy algorithm (see Section 3.5) with one clustering algorithm on one replicate on a Pentium 500.

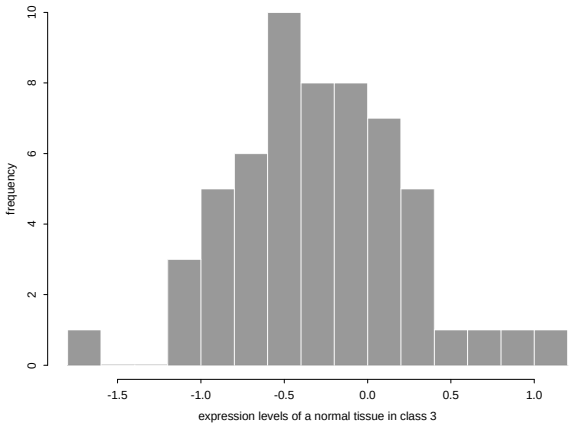

Figure 5: Histogram of the distribution of the expression levels in a normal tissue for a gene (class) in the ovary data

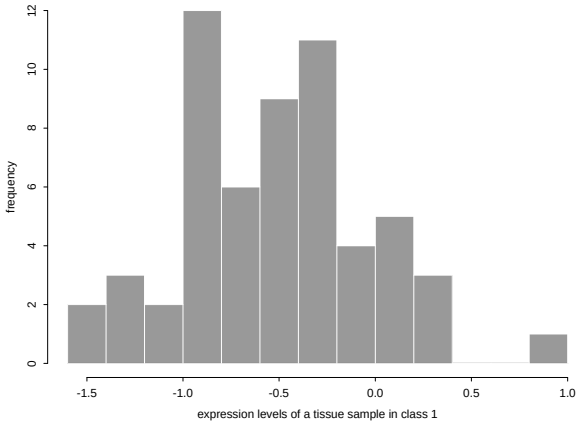

Figure 6: Histogram of the distribution of the expression levels in a tumor tissue for a gene (class) in the ovary data

Hence, the resulting sample covariance matrix of this randomly permuted data set would be close to diagonal. However, inspection of the ovary data shows that the sample covariance matrices are not too far from diagonal. Therefore, this set of randomly permuted data may be reasonable replicates of the original ovary data set.

Cyclic data with different class sizes: This synthetic data set models ten cyclic classes. The observations can be interpreted as genes and the experimental conditions can be interpreted as time points. In this data set, the cyclic behavior of genes (observations) is modeled by the $\sin$ function. Classes are modeled as genes that have similar peak times over the time series data. Different classes have different phase shifts and have different sizes.

Let $x_{i,j}$ be the simulated expression level of gene i and condition j in this simulated data set with ten classes. Let $x_{i,j} = \delta_j + \lambda_j * (\alpha_i + \beta_i \phi(i,j))$, where $\phi(i,j) = \sin(\frac{2\pi j}{8} - \frac{2\pi k}{10})$ [Zhao, 2000]. α_i represents the average expression level of gene i, which is chosen according to the normal distribution with mean 0 and standard deviation 2. β_i is the amplitude control for gene i, which is chosen according to a normal distribution with mean 3 and standard deviation 0.5. $\phi(i,j)$ models cyclic time series data. In this synthetic data set, each cycle is assumed to span 8 time points. k is the class number, which is chosen according to Zipf's Law [Zipf, 1949], which allows us to model classes with different sizes. Since different classes have different values of k, different classes are represented by different phase shifts of the $\sin$ function. λ_j is the amplitude control of condition j, chosen according to the normal distribution with mean 3 and standard deviation 0.5. δ_j represents an additive experimental error, chosen according to the standard normal distribution. Each observation (row) is normalized to have mean 0 and variance 1 before PCA or any clustering algorithm is applied.

This synthetic model suffers from the drawback that the form of the data may not be realistic for gene expression data, and the ad-hoc choice of the parameters for the distributions of α_i, β_i, λ_j, and δ_j. However, there is evidence that the form of the model is reasonable for the yeast cycle data [Zhao, 2000].

Cyclic data with spiky classes: This synthetic data has the same form as the cyclic data with different class sizes. Again, there are ten synthetic classes, and the data are generated from the equation for $x_{i,j}$. However, the class number (k in $\phi(i,j)$) is generated according to the uniform distribution, and not Zipf's law. Hence, the class sizes are approximately the same in this data. Some genes show "spiky" behavior, *i.e.,* their expression levels are changed sharply over a short period of time. We use the term *spiky classes* to refer to classes in which genes show spiky behavior. Spiky classes, are modeled by raising the $\sin$ function in $\phi(i,j)$ to higher powers. Thus, different classes are modeled by phase shifts or different "spikiness" (*i.e.,* different powers of the $\sin$ function in $\phi(i,j)$).

This synthetic data set suffers the same drawback as the cyclic data with different class sizes: the model may not be realistic, and the ad-hoc choice of the parameters for the distributions of α_i, β_i, λ_j, and δ_j. However, this synthetic data set hopes to capture a more complicated form of the real expression data by modeling classes with different shapes (*i.e.,* different spikiness) in addition to different peak times.

3.2 PCA on the data sets

We use the *covariance* matrix to define PC's for both real and synthetic data. There are two main reasons for that. First, the variables in our case are the array experiments and hence are of the same type. In particular, for the ovary data described in Section 3.1, all the experiments are scaled to have the same median prior to PCA. Second, we computed the PC's using both the correlation and the covariance matrices for the ovary and yeast cell cycle data and there is no major difference between

the two sets of PC's.

The first 14 PC's defined using the covariance matrix of the ovary data account for over 90% of the total variation of the data. On the yeast cell cycle data, the first 8 PC's defined using the covariance matrix account for over 90% of the total variation. The scree graphs for these two data sets are shown in Figures 21 and 22 in Appendix A.

3.3 Agreement between two partitions

In order to compare clustering results against external criteria, a measure of agreement is needed. Since we assume that both the external criteria and clustering results are disjoint, measures of agreement between two partitions can be used. In the statistics literature, many measures of agreement were proposed and evaluated (for example, [Rand, 1971], [Milligan *et al.*, 1983],[Hubert and Arabie, 1985], [Milligan and Cooper, 1986] and many others).

Given a set of n objects $S = \{O_1, \ldots, O_n\}$, suppose $U = \{u_1, \ldots, u_R\}$ and $V = \{v_1, \ldots, v_C\}$ represent two different partitions of the objects in S such that $\cup_{i=1}^{R} u_i = S = \cup_{j=1}^{C} v_j$ and $u_i \cap u_{i'} = \emptyset = v_j \cap v_{j'}$ for $1 \leq i \neq i' \leq R$ and $1 \leq j \neq j' \leq C$. Suppose that U is our external criterion and V is a clustering result. Let a be the number of pairs of objects that are placed in the same class in U and in the same cluster in V, b be the number of pairs of objects in the same class in U but not in the same cluster in V, c be the number of pairs of objects in the same cluster in V but not in the same class in U, and d be the number of pairs of objects in different classes and different clusters in both partitions. The quantities a and d can be interpreted as agreements, and b and c as disagreements. The Rand index [Rand, 1971] is simply $\frac{a+d}{a+b+c+d}$. The Rand index lies between 0 and 1. When the two partitions agree perfectly, the Rand index is 1.

The problem with the Rand index is that the expected value of the Rand index of two random partitions does not take a constant value (say zero). The adjusted Rand index proposed by [Hubert and Arabie, 1985] assumes the generalized hypergeometric distribution as the model of randomness, *i.e.*, the U and V partitions are picked at random such that the number of objects in the classes and clusters are fixed. Let n_{ij} be the number of objects that are in both class u_i and cluster v_j. Let $n_{i.}$ and $n_{.j}$ be the number of objects in class u_i and cluster v_j respectively. The notations are illustrated in Table 1.

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

Table 1: Notation for the contingency table for comparing two partitions.

The general form of an index with a constant expected value is $\frac{index - expected\ index}{maximum\ index - expected\ index}$, which is bounded above by 1, and takes the value 0 when the index equals its expected value.

Under the generalized hypergeometric model, it can be shown [Hubert and Arabie, 1985] that:

$$E\left[\sum_{i,j}\binom{n_{ij}}{2}\right] = \left[\sum_{i}\binom{n_{i.}}{2}\sum_{j}\binom{n_{.j}}{2}\right] / \binom{n}{2} \tag{1}$$

The expression $a + d$ can be simplified to a linear transformation of $\sum_{i,j} \binom{n_{ij}}{2}$. With simple algebra, the adjusted Rand index [Hubert and Arabie, 1985] can be simplified to:

$$\frac{\sum_{i,j} \binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i.}}{2} \sum_j \binom{n_{.j}}{2}\right] / \binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_{.j}}{2}\right] - \left[\sum_i \binom{n_{i.}}{2} \sum_j \binom{n_{.j}}{2}\right] / \binom{n}{2}} \tag{2}$$

Example 1 in Appendix B illustrates how the adjusted Rand index is computed. The Rand index for comparing the two partitions in Example 1 is 0.711, while the adjusted Rand index is 0.313. The Rand index is much higher than the adjusted Rand index, which is typical. Since the Rand index lies between 0 and 1, the expected value of the Rand index (although not a constant value) must be greater than or equal to 0. On the other hand, the expected value of the adjusted Rand index has value zero and the maximum value of the adjusted Rand index is also 1. Hence, there is a wider range of values that the adjusted Rand index can take on, thus increasing the sensitivity of the index.

In [Milligan and Cooper, 1986], many different indices were evaluated for measuring agreement between two partitions in hierarchical clustering analysis across different hierarchy levels (*i.e.*, with different numbers of clusters), and they recommended the adjusted Rand index as the index of choice. In this report, we adopt the adjusted Rand index as our measure of agreement between the external criteria and clustering results.

3.4 Clustering algorithms and similarity metrics

We implemented three clustering algorithms: the *Cluster Affinity Search Technique* (CAST) [Ben-Dor and Yakhini, 1999] the hierarchical *average-link* algorithm, and the *k-means* algorithm (with average-link initialization) [Jain and Dubes, 1988].

CAST: We implemented the pseudo-code for CAST given in [Ben-Dor and Yakhini, 1999] with two additional heuristics that have been added to BIOCLUST, the implementation of CAST by its authors. The CAST algorithm takes as input the pairwise similarities of objects and a parameter t which is a real number between 0 and 1. The parameter t is a similarity threshold to decide whether an object is added to or removed from a cluster. Thus, varying the parameter t changes the number of clusters formed. Please refer to Appendix C for more details of the algorithm.

Hierarchical average-link: Agglomerative hierarchical algorithms build clusters bottom up. Initially, each object is in its own cluster. In each step, the two clusters with the greatest cluster similarity are merged. This process is repeated until the desired number, k, of clusters is produced. In average-link, the cluster similarity criterion is the average pairwise similarity between objects in the two clusters. Refer to [Jain and Dubes, 1988] and [Anderberg, 1973] for detailed discussions on hierarchical algorithms. The average-link clustering algorithm is used by [Eisen *et al.*, 1998] to analyze gene expression data.

K-means: The number of clusters, k, is an input to the k-means clustering algorithm. Clusters are described by *centroids*, which are cluster centers, in the algorithm. In our implementation of k-means [Jain and Dubes, 1988], the initial centroids consist of the clustering results from average-link. Each object is assigned to the centroid (and hence cluster) with the closest Euclidean distance. New centroids of the k clusters are computed after all objects are assigned. The steps of assigning objects to centroids and computing new centroids are repeated until no objects are moved between clusters.

Similarity metrics: There are two popular similarity metrics used in the gene expression analysis community: Euclidean distance (for example, [Wen *et al.*, 1998]) and correlation coefficient (for

example, [Eisen *et al.*, 1998]). In our experiments, we evaluated the effectiveness of PCA on clustering analysis with both Euclidean distance and correlation coefficient, namely, CAST with correlation coefficient[4], average-link with both correlation and distance, and k-means with both correlation and distance. If Euclidean distance is used as the similarity metric, the minimum number of components in sets of PC's (m_0) considered is 2. If correlation is used, the minimum number of components (m_0) considered is 3 because there are at most 2 clusters if 2 components are used (when there are 2 components, the correlation coefficient is either 1 or -1, see Appendix D for details).

3.5 Our approach

Given a data set with an external criterion, our evaluation methodology consists of the following steps:

1. A clustering algorithm is applied to the given data set, and the adjusted Rand index with the external criterion is computed.

2. PCA is applied to the same data set. The same clustering algorithm is applied to the data after the PCA preprocessing step using the first m PC's (where $m = m_0, \ldots, p$, and m_0 is the number of components we start with, which is either 2 or 3 as explained in Section 3.4, and p is the number of experimental conditions in the data). The adjusted Rand index is computed for each of the clustering results using the first m PC's.

3. The same clustering algorithm is applied to the data after the PCA preprocessing step using different sets of PC's. The objective in this step is to find a set of PC's that gives a "high" adjusted Rand index.

One way to determine the set of PC's that gives the maximum adjusted Rand index is by exhaustive search. However, exhaustive search is very computationally intensive: for m components, there are $\binom{p}{m}$ possible sets of PC's that we have to cluster. The ovary data has 24 experimental conditions and when $m = 12$, the exhaustive search gives us $\binom{24}{12}$ (approximately 2.7 million) possible sets of PC's to cluster. Since our objective is to show that the highest adjusted Rand index is not necessarily achieved by the first m PC's, it suffices to show that there exists a set of PC's that achieves higher adjusted Rand index than the first PC's.

A simple strategy we implemented is the *greedy* approach. Let m_0 be the minimum number of components that we start with. In the greedy approach, we start with the exhaustive search for the minimum number of components, m_0. Denote the optimum set of components as S_{m_0}. For each m (where $m = (m_0 + 1), \ldots, p$), one additional component that is not already in S_{m-1} is added to the set of components, the data with all the n genes under this set of components is clustered, and the adjusted Rand index is computed. The additional component that achieves the maximum adjusted Rand index is added to S_{m-1} to form S_m. In the greedy approach, we implicitly assume that a set of PC's that achieves a high adjusted Rand index for m components is a good candidate for achieving a high adjusted Rand index for $m + 1$ components (for $m = m_0, \ldots, (p - 1)$).

Since the assumption for the greedy approach may not be satisfied, we implemented a *modified greedy* approach. The modified greedy approach requires another parameter, k, which is an integer indicating the number of "best" solutions to keep in each search step. Denote the optimum k sets of components as $\mathcal{S}_m = \{S_m^1, \ldots, S_m^k\}$, where $m = m_0, \ldots, p$. The modified greedy approach

[4]When Euclidean distance is used in CAST, the algorithm usually does not converge in practice.

also starts with an exhaustive search for the minimum number of components, m_0. However, k sets of components which achieve the top k adjusted Rand indices were stored. For each m (where $m = (m_0+1), \ldots, p$) and each of the S_m^i (where $i = 1, \ldots, k$), one additional component that is not already in S_{m-1}^i is added to the set of components, the subset of data with the extended set of components is clustered, and the adjusted Rand index is computed. The top k sets of m components that achieves the highest adjusted Rand indices are stored in $\mathcal{S}_m$. The modified greedy approach allows the search to have more choices in searching for a set of components that gives a high adjusted Rand index. Note that when $k = 1$, the modified greedy approach is identical to the simple greedy approach, and when $k = \binom{p}{m}$, the modified greedy approach is reduced to exhaustive search. So the choice for k is a tradeoff between running time and quality of solution. In our experiments, k is set to be 3.

4 Results and Discussion

We ran our experiments on two gene expression data sets and four synthetic data sets. In this section, the results of the experiments will be presented. Before the detailed results are presented for each set of experiments, here are our overall conclusions from our empirical study:

- We found that the PCA preprocessing step does not necessarily improve cluster quality, *i.e.,* the adjusted Rand indices of the clustering results on the data after PCA are not necessarily higher than the adjusted Rand indices of the clustering results on the original data on both real and synthetic data.
- We also showed that in most cases, the first m components (where $m = m_0, \ldots, p$) do not necessarily give the highest adjusted Rand index, *i.e.,* there exists another set of m components that achieves a higher adjusted Rand index than the first m components.
- There are no clear trends regarding the choice of the optimal number of PC's over all the data sets and over all the clustering algorithms and over the different similarity metrics. There is no obvious relationship between cluster quality (*i.e.,* adjusted Rand index) and the number or set of PC's used.
- In most cases, the modified greedy approach achieves higher adjusted Rand indices than the simple greedy approach.

In the following sections, the detailed experimental results on each data set is presented. For some of the results, graphs plotting the adjusted Rand index against the number of components are shown. Usually the adjusted Rand index without PCA, the adjusted Rand index of the first m components, and the adjusted Rand indices using the greedy and modified greedy approaches are shown in each graph. Note that there is only one value for the adjusted Rand index computed with the original variables (without PCA), while the adjusted Rand indices computed using PC's vary with the number of components.

4.1 Gene expression data

4.1.1 The ovary data

Figure 7 shows the result of our experiments on the ovary data using CAST [Ben-Dor and Yakhini, 1999] as the clustering algorithm and correlation coefficient as the similarity metric. The adjusted Rand in-

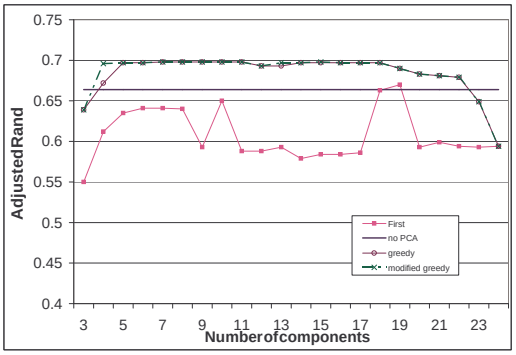

Figure 7: Adjusted Rand index against the number of components using CAST and correlation on the ovary data.

dices using the first m components (where $m = 3, \ldots, 24$) are mostly lower than that without PCA. However, the adjusted Rand indices using the greedy and modified greedy approaches for 4 to 22 components are higher than that without PCA. This shows that clustering with the first m PC's instead of the original variables may not help to extract the clusters in the data set, and that there exist sets of PC's (other than the first few which contain most of the variation in the data) that achieve higher adjusted Rand indices than without PCA. Moreover, the adjusted Rand indices computed using the greedy and modified greedy approaches are not very different for this data set using the CAST algorithm and correlation.

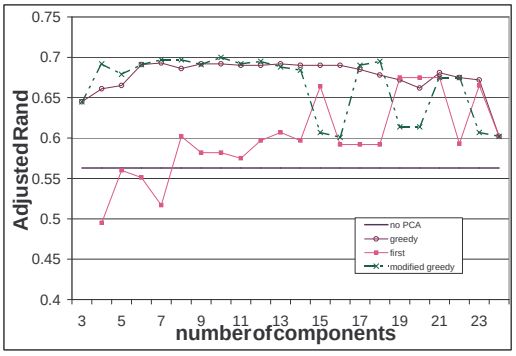

Figure 8: Adjusted Rand index against the number of components using k-means and correlation on the ovary data.

Figures 8 and 9 show the adjusted Rand indices using the k-means algorithm on the ovary data with the correlation and Euclidean distance as similarity metrics respectively. Figure 8 shows that the adjusted Rand indices using the first m components tends to increase from below the index without PCA to above that without PCA as the number of components increases. However, the results using the same algorithm but Euclidean distance as the similarity metric show a very different picture (Figure 9): the adjusted Rand indices are high for first 2 and 3 PC's and then drop drastically to below that without PCA. Manual inspection of the clustering result of the first 4 PC's using k-means and Euclidean distance shows that two classes are combined in the same cluster while the clustering result of the first 3 PC's separates the 4 classes, showing that the drastic drop in the adjusted Rand index reflects degradation of cluster quality with additional PC's. When the data points are visualized in the space of the first and second PC's, the four classes are reasonably well-separated in the Euclidean space. However, when the data points are visualized in the space of the second and fourth PC's, two classes overlap. The degradation of cluster quality with additional PC's is probably because classes are not

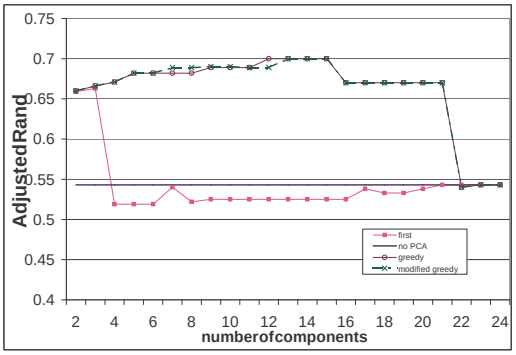

Figure 9: Adjusted Rand index against the number of components using k-means and Euclidean distance on the ovary data.

very well-separated in the Euclidean space of the higher PC's, and hence, it is more difficult for the clustering algorithms to extract the classes (see Appendix E). Figures 8 and 9 also show that different similarity metrics may have very different effect on the use of PCA as a preprocessing step to cluster analysis.

The adjusted Rand indices using the modified approach in Figure 8 show an irregular pattern. In some instances, the adjusted Rand index computed using the modified greedy approach is even lower than that using the first few components and that using the greedy approach. This shows that our heuristic assumption for the greedy approach is not always valid, *i.e.*, a set of PC's that achieve a high adjusted Rand index for m components may not be a good candidate for achieving a high adjusted Rand index for $m+1$ components (for $m = m_0, \ldots, (p-1)$). Nevertheless, the greedy and modified greedy approaches show that there exists other sets of PC's that achieve higher adjusted Rand indices than the first few PC's most of the time.

The results using the hierarchical average-link algorithm with correlation coefficient and Euclidean distance as similarity metrics show a similar pattern to the results using k-means (graphs not shown here).

Note that the adjusted Rand index without PCA using CAST and correlation (0.664) is much higher than that using k-means (0.563) and average-link (0.572) using the same similarity metric. Manual inspection of the clustering results without PCA shows that only CAST clusters mostly contain clones from each class, while k-means and average-link clustering results combine two classes into one cluster. This confirms that higher adjusted Rand indices reflect higher cluster quality with respect to the external criteria. With the first m components, CAST with correlation has a similar range of adjusted Rand indices to the other algorithms (approximately between 0.55 to 0.68). The rule of thumb of choosing the first 14 PC's to cover 90% of the total variation in the data would have a detrimental effect on cluster quality if CAST with correlation, k-means with distance, or average-link with distance is the algorithm being used.

When correlation is used (Figures 7 and 8), the adjusted Rand index using all 24 PC's is not the same as that using the original variables. On the other hand, when Euclidean distance is used (Figure 9), the adjusted Rand index using all 24 PC's is the same as that with the original variables. This is because the Euclidean distance between a pair of genes using all the PC's is the same as that using the original variables. But correlation coefficient is not preserved after PCA (Section 2.5).

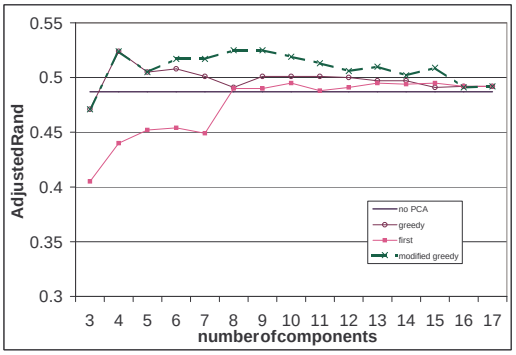

Figure 10: Adjusted Rand index against the number of components using CAST and correlation on the yeast cell cycle data.

4.1.2 The yeast cell cycle data

Figure 10 shows the result on the yeast cell cycle data using CAST [Ben-Dor and Yakhini, 1999] as the clustering algorithm and correlation coefficient as the similarity metric. The adjusted Rand indices using the first 3 to 7 components are lower than that without PCA, while the adjusted Rand indices with the first 8 to 17 components are comparable to that without PCA.

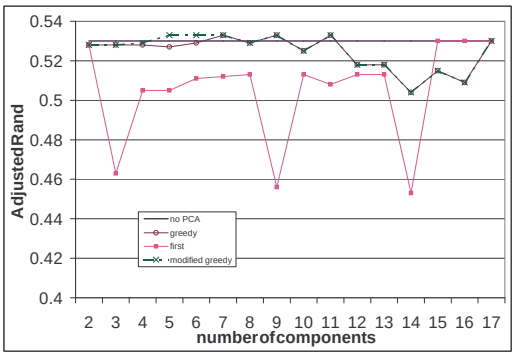

Figure 11: Adjusted Rand index against the number of components using k-means and Euclidean distance on the yeast cell cycle data.

Figure 11 shows the result on the yeast cell cycle data using k-means and Euclidean distance. The adjusted Rand indices without PCA are relatively high compared to those using PC's. Figure 11 on the yeast cell cycle data shows a very different picture than Figure 9 on the ovary data. This shows that the effectiveness of clustering with PC's depends on the data set being used.

The results on the yeast cell cycle data sets using k-means with correlation and average-link (with both correlation and Euclidean distance) are not shown here.

4.2 Synthetic data

4.2.1 Mixture of normal distributions on the ovary data

Figure 12 shows the results of our experiments on the synthetic mixture of normal distributions on the ovary data using CAST [Ben-Dor and Yakhini, 1999] as the clustering algorithm and correlation coefficient as the similarity metric. The lines in Figure 12 represent the average adjusted Rand indices over the 10 replicates of the synthetic data, and the error bars represent one standard deviation from the mean for the modified greedy approach and for using the first m PC's. The error bars show that

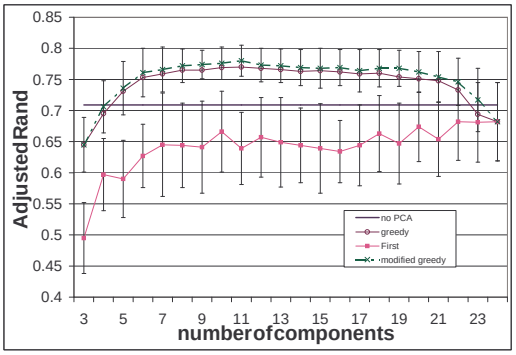

Figure 12: Adjusted Rand index against the number of components using CAST and correlation on the mixture of normal distributions on the ovary data.

the standard deviations using the modified greedy approach tend to be lower than that using the first m components. A careful study also shows that the modified greedy approach has lower standard deviations than the greedy approach (data not shown here). This shows that the modified greedy approach is more robust than the greedy approach in identifying a set of components with a high adjusted Rand index. The error bars for the case without PCA are not shown for clarity of the figure. The standard deviation for the case without PCA is 0.064 for this set of synthetic data, which would overlap with those using the first components and the modified greedy approach. A manual study of the experimental results from each of the 10 replicates (details not shown here) shows that 8 out of the 10 replicates show very similar patterns to the average pattern in Figure 12, *i.e.*, most of the cluster results with the first m components have lower adjusted Rand indices than that without PCA, and the results using the greedy and modified greedy approach are slightly higher than that without PCA. In the following results, only the average patterns will be shown.

Figure 12 shows a similar trend to real data in Figure 7, but the synthetic data has higher adjusted Rand indices for the clustering results without PCA and with the greedy and modified greedy approaches. As in the case with real data (Figure 7), the adjusted Rand indices with the first PC's lie below that without PCA, and those with the greedy and modified greedy approach for 4 to 22 components are above that without PCA.

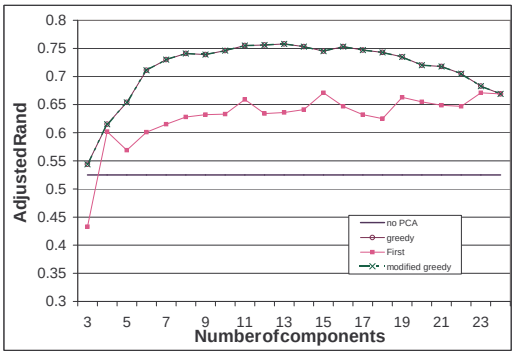

Figure 13: Average adjusted Rand index against the number of components using k-means and correlation on the mixture of normal distributions of the ovary data.

The average adjusted Rand indices using the k-means algorithm with the correlation and Euclidean distance as similarity metrics are shown in Figure 13 and Figure 14 respectively. In Figure 13, the adjusted Rand indices using the first m components gradually increase as the number of components increases, while in Figure 14, the adjusted Rand indices using the first m indices are mostly below that

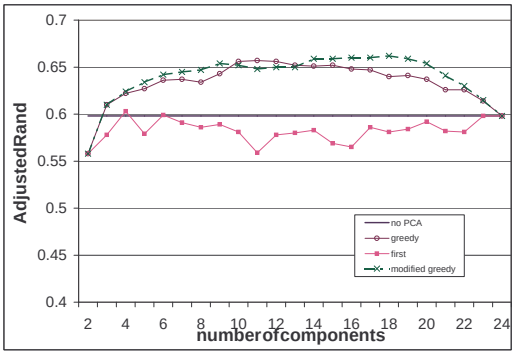

Figure 14: Average adjusted Rand index against the number of components using k-means and Euclidean distance on the mixture of normal distributions of the ovary data.

without PCA.

The results using average-link and correlation (not shown here) are similar to that of k-means and correlation. The average adjusted Rand indices using average-link and Euclidean distance (not shown here) tend to decrease as the number of components is increased.

4.2.2 Randomly permuted ovary data

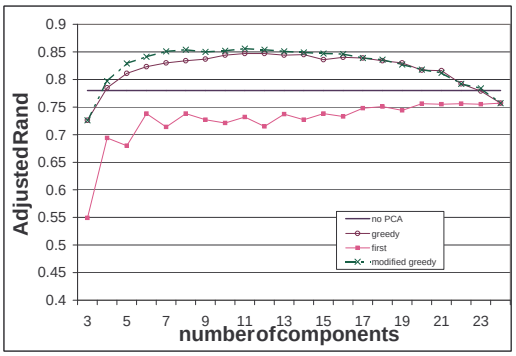

Figure 15: Average adjusted Rand index against the number of components using CAST and correlation on the randomly permuted ovary data.

Figures 15, 16, and 17 show the average adjusted Rand indices using CAST and correlation, k-means with correlation, and k-means with Euclidean distance on the randomly permuted ovary data respectively. The general trend is very similar to the results on the mixture of normal distributions in Section 4.2.1. The average adjusted Rand indices computed from CAST clusters in Figure 15 using the first m PC's lie below that using the original variables (similar to Figure 12). When the k-means algorithm is used with correlation as the similarity metric in Figure 16, the average adjusted Rand indices using the first m PC's tend to increase as the number of components increases (similar to Figure 13). When the modified greedy or the greedy approach is used, the average adjusted Rand indices for all clustering results (except CAST with 3 components) with all of the algorithms are above that without PCA.

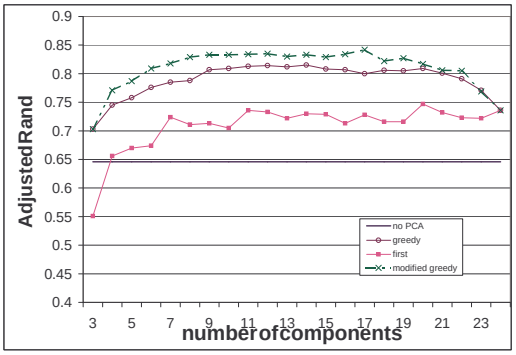

Figure 16: Average adjusted Rand index against the number of components using k-means and correlation on the randomly permuted ovary data.

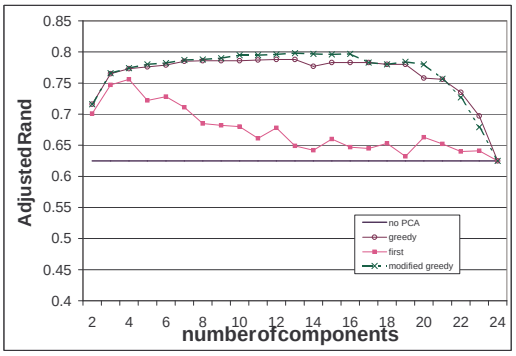

Figure 17: Average adjusted Rand index against the number of components using k-means and Euclidean distance on the randomly permuted ovary data.

4.2.3 Cyclic data with different cluster sizes

The results using this synthetic data set are very different than those using the mixture of normal distributions and the randomly permuted ovary data. The average adjusted Rand index without PCA (0.941) is very high. The high adjusted Rand index indicates that the data set is very clean and the clustering algorithms recover most of the classes.

Figure 18 shows the average adjusted Rand indices using CAST and correlation. Manual inspection shows that in 7 out of the 10 replicates, the adjusted Rand index without PCA is perfect (*i.e.*, 1). The average adjusted Rand indices using the first m components are much lower than that without PCA in Figure 18. Note that there is sharp decline in the average adjusted Rand index when all 24 components are used in the greedy approach in Figure 18. This is no accident. A close inspection shows that 9 out of the 10 replicates show a drastic decline in adjusted Rand index when all the components are used with the greedy approach. In all of the 9 replicates, the additional component which contributes to the sharp decline in the adjusted Rand index is the third PC. The results of the modified greedy approach are not shown since the greedy approach achieves almost perfect adjusted Rand indices.

Figure 19 shows the average adjusted Rand indices with the k-means algorithm and Euclidean distance as the similarity metric. Again, the adjusted Rand index without PCA is very high. But in this case, using the first m components achieve higher or comparable adjusted Rand indices to that without PCA.

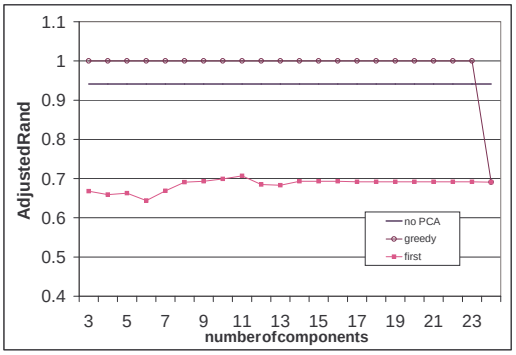

Figure 18: Average adjusted Rand index against the number of components using CAST and correlation on the cyclic data with different cluster sizes.

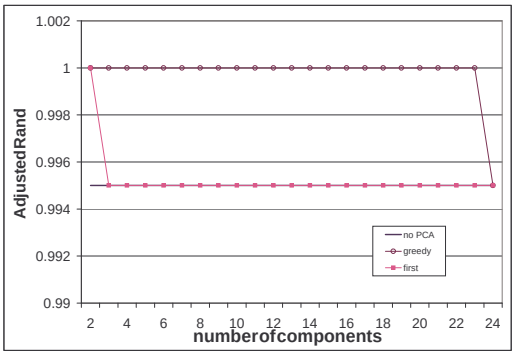

Figure 19: Average adjusted Rand index against the number of components using k-means and Euclidean distance on the cyclic data with different cluster sizes.

4.2.4 Cyclic data with spiky clusters

The general trend using this set of synthetic data sets is very different than the other synthetic data sets (even the data set in Section 4.2.3): using PCA (with the first components or the greedy or modified greedy approach) helps to achieve higher adjusted Rand indices.

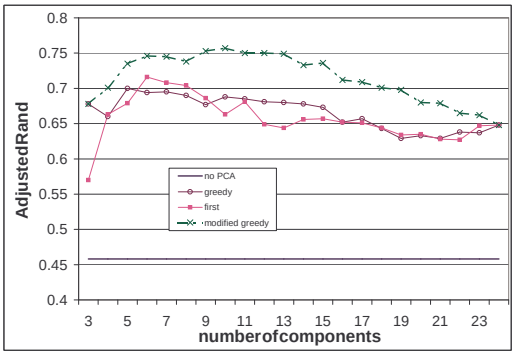

Figure 20: Average adjusted Rand index against the number of components using CAST and correlation on the cyclic data with spiky clusters.

Figure 20 shows an example of results applying CAST and correlation to this set of synthetic data. Unlike the results with other real or synthetic data sets, the adjusted Rand indices of CAST clusters with the first m PC's lie above that without PCA in Figure 20. The results using other algorithms also show a similar trend.

5 Conclusions and Future Work

Our experiments on two real gene expression data sets and four sets of synthetic data show that clustering with the PC's instead of the original variables does not necessarily improve cluster quality. Our empirical study shows that the traditional wisdom that the first few PC's that contain most of the variation in the data may help to extract cluster structure is generally *not* true. We also show that there usually exists some sets of m PC's that achieve higher quality of clustering results than the first m PC's.

Our empirical results show that clustering with PC's has different impact on different algorithms and different similarity metrics. When CAST is used with correlation as the similarity metric, clustering with the first m components usually gives a lower adjusted Rand index than clustering with the original variables (this is true in both of the real gene expression data sets and in 3 out of the 4 synthetic data sets). On the other hand, when k-means is used with correlation as the similarity metric, using *all* of the PC's in cluster analysis instead of the original variables gives higher or similar adjusted Rand indices on all of our real and synthetic data sets. When Euclidean distance is used as the similarity metric, clustering (either with k-means or average-link) using the first couple of PC's usually achieves higher or comparable adjusted Rand indices to without PCA, but the adjusted Rand indices drop sharply with more PC's. Since the Euclidean distance computed with the first m PC's is just an approximation to the Euclidean distance computed with all the experiments, the first couple of PC's probably contain most of the cluster information while the last PC's are mostly noise. There is no clear indication from our results what should be the number of PC's to use in the case of Euclidean distance. Using the number of first PC's chosen by the rule of thumb to cover 90% of the total variation in the data is too many in the case of Euclidean distance on the ovary data and yeast cell cycle data. Based on our empirical results, we recommend against using the first few PC's if CAST and correlation is used to cluster a gene expression data set. On the other hand, we recommend using all of the PC's if k-means and correlation is used instead. However, the increased adjusted Rand indices using the "appropriate" PC's with k-means and average-link are comparable to that of CAST using the original variables in many of our results. Therefore, choosing a good clustering algorithm is as important as choosing the "appropriate" PC's.

There does not seem to be any general relationship between cluster quality (*i.e.,* adjusted Rand index with an external standard) and the number of components used based on the results on both real and synthetic data sets. The choice of the first few components is usually not optimal (except when Euclidean distance is used), and usually may even achieve lower adjusted Rand indices than without PCA. There usually exists another set of PC's that achieves higher adjusted Rand indices than clustering with the original variables or with the first m PC's. However, there does not seem to be any general trend for the the set of components chosen by the greedy or modified greedy approach that has a high adjusted Rand index. Usually, there are no external criteria available for real gene expression data, so it would be very useful if a rule to choose PC's for cluster analysis is available. A careful manual inspection of our empirical results shows that the first two PC's are usually chosen in the exhaustive search step for the set of m_0 components that give the highest adjusted Rand indices. In fact, when CAST is used with correlation as the similarity metric, the 3 components found in the exhaustive search step *always* include the first two PC's on *all* of our real and synthetic data sets. The first two PC's are *usually* returned by the exhaustive search step when k-means with correlation, or k-means with Euclidean distance, or average-link with correlation is used on all of our data sets except the synthetic data set with spiky clusters. On our real gene expression data sets, the first 2 PC's are always returned by the exhaustive search step except when k-means with Euclidean distance is applied

to the ovary data.

The patterns of results using the synthetic mixture of normal distributions and the randomly permuted ovary data are very similar to those on the actual ovary data. This implies that these two synthetic data sets probably have similar complexities as the real gene expression data set. On the other hand, the patterns of results on both cyclic data sets are very different than those on the yeast cell cycle and those on the ovary data, implying that the cyclic data sets may not be as satisfactory models for synthesizing gene expression data as the mixture of normal and the randomly permuted models. In particular, the cyclic data set with different cluster sizes achieves very high (close to 1) adjusted Rand indices even without PCA, which is very different than what we observe on real data.

There are a few possible directions of both empirical and theoretical future work. Empirically, it would be interesting to generate more replicates (*i.e.,* more than 10) for each set of synthetic data to see if the standard deviations from the average pattern would go down. Furthermore, it would be interesting to check if increasing the parameter k (the number of best sets of components to keep in each search step) would significantly improve the adjusted Rand index computed. Comparing the adjusted Rand indices using a set of random PC's to those computed with the greedy approach would also be interesting. Our observation above suggests that the first two PC's are usually chosen in the exhaustive search step. Generating a set of random PC's that always includes the first two PC's, and then applying clustering algorithms and computing the adjusted Rand indices may also lead to interesting insights. In terms of future theoretical work, it is interesting to develop time-efficient approximation algorithms (other than the greedy and modified greedy approaches) to compute a set of PC's that achieves a high adjusted Rand index. Developing other models for generating synthetic gene expression data is definitely of interest. In addition, formally testing the normality of the ovary data would be useful to support the synthetic data model of the mixture of multivariate normal distributions.

Our empirical study shows that the effectiveness of PCA on cluster analysis depends on the particular data set, the clustering algorithm and the similarity metric used. For most real gene expression data sets, an external criterion to assess clustering results is not available. It would be very valuable to develop a methodology that does not require an external criterion to evaluate the effectiveness of PCA as a preprocessing step. In our previous work [Yeung *et al.*, 2000], we proposed a methodology that estimates the predictive power of clustering algorithms. We believe that our methodology in [Yeung *et al.*, 2000] can be modified to investigate the effectiveness of clustering with the PC's instead of the original variables. It would be interesting to compare the evaluation results using our methodology with the results in this paper using external criteria.

To conclude, we believe that our empirical study is one step forward to investigate the effectiveness of clustering with the PC's instead of the original variables.

Acknowledgement

I would like to thank Michel Schummer from the Department of Molecular Biotechnology at University of Washington for his unpublished ovary data set and his feedback. I would also like to thank Chris Frayley from the Statistics Department at University of Washington for sharing with me the Splus code to generate a mixture of Gaussian variables. I would also like to thank Lue Ping Zhao at the Fred Hutchinson Cancer Research Center for his ideas of the cyclic models for gene expression data. I would also like to thank Amir Ben-Dor at Agilent Laboratories for sharing the additional heuristics implemented in BIOCLUST with me. In addition, I would like to thank my friends Mathieu Blanchette, Jeremy Buhler and Don Patterson for their comments on an early draft of this paper. Finally, I would like to thank my advisor Larry Ruzzo, and my committee members Mark Campbell, Pedro Domingos, Phil Green, David Haynor and Martin Tompa for their feedback and suggestions during the examina-

tion and for the writeup.

Appendix

A Scree graphs for the ovary and yeast cell cycle data

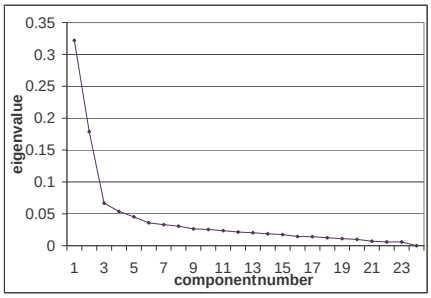

Figure 21: Scree graph for the ovary data

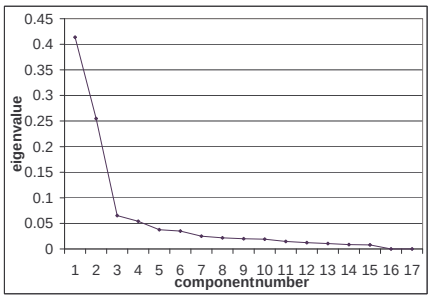

Figure 22: Scree graph for the yeast cell cycle data

Figure 21 shows the scree graph for the ovary data. There is sharp change of steepness in Figure 21 at 3 components, and another gentle change at 6 components. Similarly, Figure 22 shows the scree graph for the yeast cell cycle data. Again, there is a sharp change of steepness at 3 components, and another gentle change at 5 components. These two examples illustrate that the scree graph approach to decide the number of first PC's to be used is very ad-hoc and subjective.

B Example illustrating the adjusted Rand index

The following example illustrates how the adjusted Rand index (discussed in Section 3.3) is computed. Example 1 is a contingency table in the same form as in Table 1.

a is defined as the number of pairs of objects in the same class in U and same cluster in V, hence a can be written as $\sum_{i,j} \binom{n_{ij}}{2}$. In Example 1, $a = \binom{2}{2} + \binom{4}{2} = 7$. b is defined as the number of pairs of objects in the same class in U but not in the same cluster in V. In terms of the notation in Table 1, b can be written as $\sum_i \binom{n_{i.}}{2} - \sum_{i,j} \binom{n_{ij}}{2}$ In Example 1, $b = \binom{2}{2} + \binom{4}{2} + \binom{4}{2} - 7 = 6$. Similarly, c is defined as the number of pairs of objects in the same cluster in V but not in the same

Class or Cluster	v_1	v_2	v_3	*Sums*
u_1	1	1	0	2
u_2	1	2	1	4
u_3	0	0	4	4
Sums	2	3	5	$n = 10$

Example 1

class in U, so c can be written as $\sum_j \binom{n_{.j}}{2} - \sum_{i,j} \binom{n_{ij}}{2} = \binom{2}{2} + \binom{3}{2} + \binom{5}{2} - 7 = 7$. d is defined as the number of pairs of objects that are not in the same class in U and not in the same cluster in V. Since $a + b + c + d = \binom{n}{2}$, $d = \binom{10}{2} - 7 - 6 - 7 = 25$. The Rand index for comparing the two partitions in Example 1 is $\frac{7+25}{45} = 0.711$, while the adjusted Rand index is $\frac{7-14*13/45}{(14+13)/2-14*13/45} = 0.313$ (see Section 3.3 for the definitions of the Rand and adjusted Rand indices). The Rand index is much higher than the adjusted Rand index.

C Details of the CAST algorithm

The Cluster Affinity Search Technique (CAST) is an algorithm proposed by [Ben-Dor and Yakhini, 1999] to cluster gene expression data. The input to the algorithm includes the pairwise similarities of the genes, and a cutoff parameter t (which is a real number between 0 and 1). The clusters are constructed one at a time. The current cluster under construction is called C_{open}. The *affinity* of a gene g, $a(g)$, is defined to be the sum of similarity values between g and all the genes in C_{open}. A gene g is said to have high affinity if $a(g) \geq t|C_{open}|$. Otherwise, g is said to have low affinity. Note that the affinity of a gene depends on the genes that are already in C_{open}. The algorithm alternates between adding high affinity genes to C_{open}, and removing low affinity genes from C_{open}. C_{open} is *closed* when no more genes can be added to or removed from it. Once a cluster is closed, it is not considered any more by the algorithm. The algorithm iterates until all the genes have been assigned to clusters and the current C_{open} is closed.

When a new cluster C_{open} is started, the initial affinity of all genes are 0 since C_{open} is empty. One additional heuristic that the authors [Ben-Dor and Yakhini, 1999] implemented in their software BIOCLUST is to choose a gene with the maximum number of neighbors to start a new cluster. Another heuristic is that after the CAST algorithm converges, there is an additional iterative step, in which all clusters are considered at the same time, and genes are moved to the cluster with the highest average similarity.

D Correlation coefficient when there are 2 components

When there are 2 components, the correlation coefficient is either 1 or -1. Suppose there are two genes g_1 and g_2 with two components. Let $x_{i,j}$ (where $i, j = 1, 2$) be the expression level of gene i under component j. The correlation coefficient between g_1 and g_2 can be simplified to:

$$\frac{(x_{1,1} - x_{1,2}) * (x_{2,1} - x_{2,2})}{\sqrt{(x_{1,1} - x_{1,2})^2 * (x_{2,1} - x_{2,2})^2}} \tag{3}$$

Since the denominator in Equation 3 represents the product of the norms of genes g_1 and g_2, the denominator must be positive. From Equation 3, the correlation coefficient between genes g_1 and g_2 is 1 if $(x_{1,1} - x_{1,2}) * (x_{2,1} - x_{2,2}) > 0$, the correlation coefficient is -1 if $(x_{1,1} - x_{1,2}) * (x_{2,1} - x_{2,2}) < 0$. If $x_{1,1} = x_{1,2}$ or $x_{2,1} = x_{2,2}$, the correlation coefficient is undefined. Since there are only two possible values that the correlation coefficient can take when there are two components, there are at most two clusters.

E Visualization of the clustering result with k-means and Euclidean distance on the ovary data

The results on the ovary data set using k-means and Euclidean distance in Figure 9 show that the adjusted Rand indices are high for first 2 and 3 PC's and then drop drastically to below that without PCA. When the four classes of the ovary data are viewed in the space of the first two PC's (Figure 23), the four classes are reasonably well-separated in the Euclidean space. In fact, when the clustering result using the first 3 PC's is viewed in the space of the first two PC's, the four clusters correspond mostly to the four classes in Figure 24. However, two classes (class 1 and class 4) overlap in Figure 25, in which the classes are viewed in the space of the second and fourth PC's. In fact, manual inspection shows that the two overlapping classes were combined into one cluster by k-means with Euclidean distance when four PC's are used. The fourth PC probably is mostly noise, which makes it more difficult for k-means to extract the four classes.

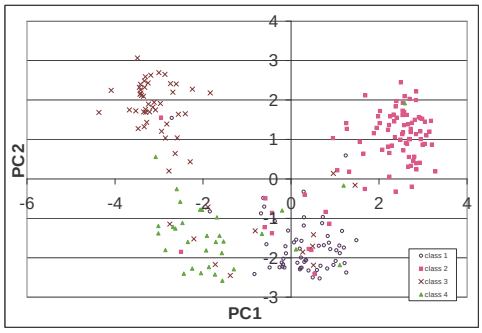

Figure 23: Visualization of the four classes from the ovary data in the space of the first two PC's.

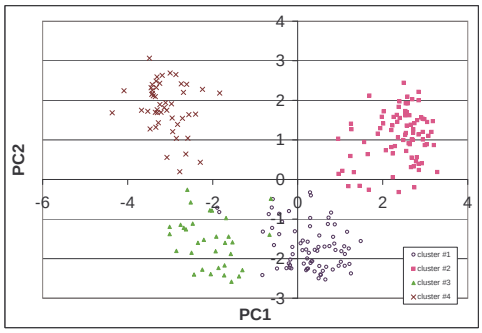

Figure 24: Visualization of the four clusters using the first 3 PC's and k-means with Euclidean distance from the ovary data in the space of the first two PC's.

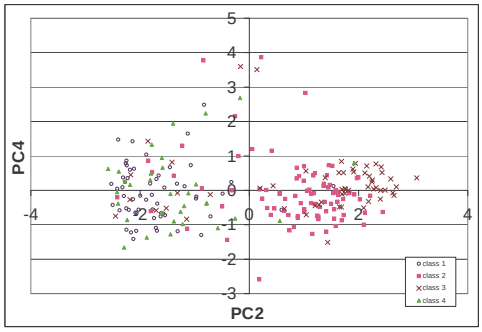

Figure 25: Visualization of the four classes from the ovary data in the space of the second and fourth PC's.

References

[Anderberg, 1973] Anderberg, M. R. (1973) *Cluster analysis for applications*. Academic Press.

[Ben-Dor and Yakhini, 1999] Ben-Dor, A. and Yakhini, Z. (1999) Clustering gene expression patterns. In *RECOMB99: Proceedings of the Third Annual International Conference on Computational Molecular Biology*, Lyon, France.

[Chang, 1983] Chang, W. C. (1983) On using principal components before separating a mixture of two multivariate normal distributions. *Applied Statistics*, **32**, 267–275.

[Chen *et al.*, 1999] Chen, T., Filkov, V. and Skiena, S. S. (1999) Identifying gene regulatory networks from experimental data. In *RECOMB99: Proceedings of the Third Annual International Conference on Computational Molecular Biology*, Lyon, France.

[Cho *et al.*, 1998] Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J. and Davis, R. W. (1998) A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular Cell*, **2**, 65–73.

[Chu *et al.*, 1998] Chu, S., DeRisi, J., Eisen, M., Mulholland, J., Botstein, D., Brown, P. O. and Herskowitz, I. (1998) The transcriptional program of sporulation in budding yeast. *Science*, **282**, 699–705.

[Dunteman, 1989] Dunteman, G. H. (1989) *Principapl Components Analysis*. Sage Publications.

[Eisen *et al.*, 1998] Eisen, M. B., Spellman, P. T., Brown, P. O. and Botstein, D. (1998) Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Science USA*, **95**, 14863–14868.

[Everitt and Dunn, 1992] Everitt, B. S. and Dunn, G. (1992) *Applied Multivariate Data Analysis*. Oxford University Press.

[Hubert and Arabie, 1985] Hubert, L. and Arabie, P. (1985) Comparing partitions. *Journal of Classification*, 193–218.

[Jain and Dubes, 1988] Jain, A. K. and Dubes, R. C. (1988) *Algorithms for Clustering Data*. Prentice Hall, Englewood Cliffs, NJ.

[Jolliffe, 1986] Jolliffe, I. T. (1986) *Principal Component Analysis*. New York : Springer-Verlag.

[Jolliffe *et al.*, 1980] Jolliffe, I. T., Jones, B. and t. Morgan, B. J. (1980) cluster analysis of the elderly at home: a case study. *Data analysis and Informatics*, 745–757.

[Lander, 1999] Lander, E. S. (1999) Array of hope. *Nature Genetics*, **21**, 3–4.

[Milligan and Cooper, 1986] Milligan, G. W. and Cooper, M. C. (1986) A study of the comparability of external criteria for hierarchical cluster analysis. *Multivariate Behavioral Research*, **21**, 441–458.

[Milligan *et al.*, 1983] Milligan, G. W., Soon, S. C. and Sokol, L. M. (1983) The effect of cluster size, dimensionality, and the number of clusters on recovery of true cluster structure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **5**, 40–47.

[Rand, 1971] Rand, W. M. (1971) Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, **66**, 846–850.

[Raychaudhuri *et al.*, 2000] Raychaudhuri, S., Stuart, J. M. and Altman, R. B. (2000) Principal components analysis to summarize microarray experiments: application to sporulation time series. In *Pacific Symposium on Biocomputing*, vol. 5.

[Schummer, 2000] Schummer, M. (2000) Manuscript in preparation.

[Schummer *et al.*, 1999] Schummer, M., Ng, W. V., Bumgarner, R. E., Nelson, P. S., Schummer, B., Bednarski, D. W., Hassell, L., Baldwin, R. L., Karlan, B. Y. and Hood, L. (1999) Compartive hybridization of an array of 21500 ovarian cdnas for the discovery of genes overexpressed in ovarian carcinomas. *An International Journal on Genes and Genomes*, **238**, 375–385.

[Tamayo *et al.*, 1999] Tamayo, P., Slonim, D., Mesirov, J., Zhu, Q., Kitareewan, S., Dmitrovsky, E., Lander, E. S. and Golub, T. R. (1999) Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Science USA*, **96**, 2907–2912.

[Wen *et al.*, 1998] Wen, X., Fuhrman, S., Michaels, G. S., Carr, D. B., Smith, S., Barker, J. L. and Somogyi, R. (1998) Large-scale temporal gene expression mapping of central nervous system development. *Proceedings of the National Academy of Science USA*, **95**, 334–339.

[Yeung *et al.*, 2000] Yeung, K. Y., Haynor, D. R. and Ruzzo, W. L. (2000) Validating clustering for gene expression data. Tech. Rep. UW-CSE-00-01-01, Dept. of Computer Science and Engineering, University of Washington.

[Zhao, 2000] Zhao, L. P. (2000) Personal communications.

[Zipf, 1949] Zipf, G. K. (1949) *Human behavior and the principle of least effort*. Addison-Wesley.