LETTER Communicated by Joachim Buhmann

Resampling Method for Unsupervised Estimation of Cluster Validity

Erel Levine
Eytan Domany
Department of Physics of Complex Systems, Weizmann Institute of Science, Rehovot 76100, Israel

We introduce a method for validation of results obtained by clustering analysis of data. The method is based on resampling the available data. A figure of merit that measures the stability of clustering solutions against resampling is introduced. Clusters that are stable against resampling give rise to local maxima of this figure of merit. This is presented first for a one-dimensional data set, for which an analytic approximation for the figure of merit is derived and compared with numerical measurements. Next, the applicability of the method is demonstrated for higher-dimensional data, including gene microarray expression data.

1 Introduction

Cluster analysis is an important tool for investigating and interpreting data. Clustering techniques are the main tool used for exploratory data analysis, when one is dealing with data about whose internal structure little or no prior information is available. Cluster algorithms are expected to produce partitions that reflect the internal structure of the data and identify "natural" classes and hierarchies present in it.

A wide variety of clustering algorithms have been proposed. Some have their origins in graph theory, whereas others are based on statistical pattern recognition, self-organization methods, and more. More recently, algorithms rooted in statistical mechanics have been introduced.

Comparing the relative merits of various methods is made difficult by the fact that when applied to the same data set, different clustering algorithms often lead to markedly different results. In some cases such differences are expected, since different algorithms make different (explicit or implicit) assumptions about the structure of the data. If the particular set that is being studied consists, for example, of several clouds of data point, with each cloud spherically distributed about its center, methods that assume such structure (e.g., k-means) will work well. On the other hand, if the data consist of a single nonspherical cluster, the same algorithms will fare miserably, breaking it up into a hierarchy of partitions. Since for the cases of interest one does not know which assumptions are satisfied by the data, a researcher

may run into severe difficulties in interpreting results. By preferring one algorithm's clusters over those of another, he or she may reintroduce his or her biases about the underlying structure—precisely those biases that he or she hoped to eliminate by employing clustering techniques. In addition, the differences in the sensitivity of different algorithms to noise, which inherently exists in the data, also yield a major contribution to the difference between their results.

The ambiguity is made even more severe by the fact that even when one sticks exclusively to one's favorite algorithm, the results may depend strongly on the values assigned to various parameters of the particular algorithm. For example, if there is a parameter that controls the resolution at which the data are viewed, the algorithm produces a hierarchy of clusters (a dendrogram) as a function of this parameter. One then has to decide which level of the dendrogram reflects best the "natural" classes present in the data.

Needless to say, one wishes to answer these questions in an unsupervised manner, making use of nothing more than the available data. Various methods and indicators that come under the name "cluster validation" attempt to evaluate the results of cluster analysis in this manner (Jain & Dubes, 1988). Numerous studies suggest direct and indirect indices for evaluation of hard clustering (Jain & Dubes, 1988; Bock, 1985), probabilistic clustering (Duda & Hart, 1973), and fuzzy clustering (Windham, 1982; Pal & Bezdek, 1995) results. Hard clustering indices are often based on some geometrical motivation to estimate how compact and well separated the clusters are; an example is Dunn's index (Dunn, 1974) and its generalizations (Bezdek & Pal, 1995); others are statistically motivated, for example, comparing the within-cluster scattering with the between-cluster separation (Davies & Bouldin, 1979). Probabilistic and fuzzy indices are not considered here. Indices proposed for these methods are based on likelihood-ratio tests (Duda & Hart, 1973), information-based criteria (Cutler & Windham, 1994), and more.

Another approach to cluster validity includes some variant of cross-validation (Fukunaga, 1990). Such methods were introduced in the context of both hard clustering (Jain & Moreau, 1986) and fuzzy clustering (Smyth, 1996). The approach presented here falls into this category.

In this article, we present a method to help select which clustering result is more reliable. The method can be used to compare different algorithms, but it is most suitable to identify, within the same algorithm, those partitions that can be attributed to the presence of noise. In these cases, a slight modification of the noise may alter the cluster structure significantly. Our method controls and alters the noise by means of resampling the original data set.

In order to illustrate the problem we wish to address and its proposed solution, consider the following example. A scientist is investigating mice and comes to suspect that there are several types of them. She therefore measures two features of the mice (such as weight and shade), looks for clusters in this two-dimensional data set, and indeed finds two clusters.

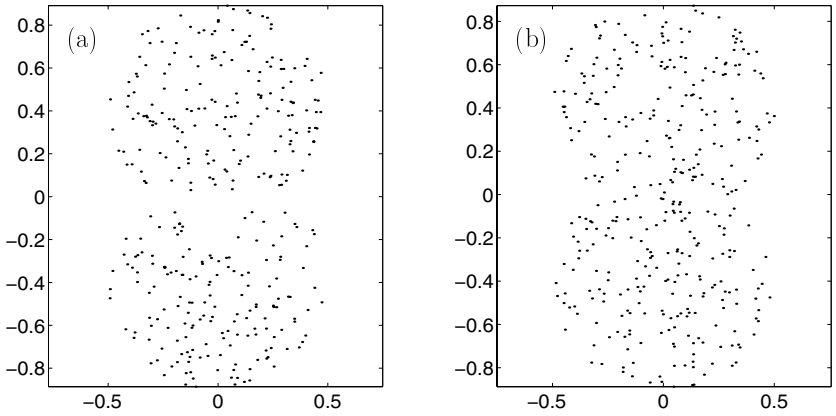

Figure 1: Two samples, drawn from an eight-shaped uniform distribution. Sample *a* is somewhat nontypical.

She can therefore conclude that there are two types of mice in her lab. Or can she?

Imagine that the data that the scientist collects can be represented by the points on Figure 1a. This data set was in fact taken from a shaped uniform distribution (with a relatively narrow "neck" at the middle). Hence, no partition really exists in the underlying structure of the data, and unless one makes explicit assumptions about the shape of the data clouds, one should identify a single cluster. The particular cluster algorithm used breaks the data into two clusters along the horizontal gap seen in Figure 1a. This gap happens to be the result of fluctuations in the data or noise in the sampling (or measurement) process. More typical data sets, such as that of Figure 1b, do not have such gaps and are not broken into two clusters by the algorithm.

If more than a single sample had been available, it would have been safe to assume that this particular gap would not have appeared in most samples. The partition into two clusters in that case would be easily identified as unreliable.[1] In most cases, however, only a single sample is available, and resampling techniques are needed in order to generate several ones.

In this article we propose a cluster validation method based on resampling (Good, 1999; Efron & Tibshirani, 1993): subsets of the data under investigation are constructed randomly, and the cluster algorithm is applied to each subset. The resampling scheme is introduced in section 2, and a figure of merit is proposed to identify the stable clustering solutions, which are

[1] By "unreliable" we mean that one cannot infer whether the partition into two clusters is only the consequence of noise and therefore cannot answer the question of whether it is likely to reappear in another random sample.

less likely to be the results of noise or fluctuations. The proposed procedure is tested in section 3 on a one-dimensional data set, for which an analytical expression for the figure of merit is derived and compared with the corresponding numerical results. In section 4 we demonstrate the applicability of our method to two artificial data sets (in $d = 2$ dimensions) and to real (very high-dimensional) DNA microarray data.

2 The Resampling Scheme

Let us denote the number of data points in the set to be studied by N. Typically, application of any cluster algorithm necessitates choosing specific values for some parameters. The results yielded by the clustering algorithm may depend strongly on this choice. For example, some algorithms (e.g., the c-shell fuzzy-clustering in Bezdek, 1981, and the iso-data algorithm in Cover & Thomas, 1991) take the expected number of clusters as part of their input. Other algorithms (e.g., Valley-Seeking, of Fukunaga, 1990) have the number of neighbors of each point as an external parameter.

In particular, many algorithms have a parameter that controls the resolution at which clusters are identified. In agglomerative clustering methods, for example, this parameter defines the level of the resulting dendrogram at which the clustering solution is identified (Jain & Dubes, 1988). For the K-nearest-neighbor algorithm (Fukunaga, 1990), a change in the number of neighbors of each point, K, controls the resolution. For deterministic annealing (DA) (Rose, Gurewitz, & Fox, 1990) and the superparamagnetic clustering algorithm (SPC) (Blatt, Wiseman, & Domany, 1996; Domany, 1999), this role is played by the temperature T.

As this control parameter is varied, the data points get assigned to different clusters, giving rise to a hierarchy. At the lowest resolution, all N points belong to one cluster, whereas at the other extreme, one has N clusters, of a single point in each. As the resolution parameter varies, clusters of data points break into subclusters, which break further at a higher level. Sometimes the aim is to generate precisely such a dendrogram. In other cases, one would like to produce a single partitioning of the data, which captures a particular important aspect. In such a case, we wish to identify that value of the resolution control parameter at which the most reliable, natural clusters appear. In these situations, the resolution parameter plays the role of one (probably the most important) member of the family of parameters of the algorithm that needs to be fixed. Let us denote the full set of parameters of our algorithm by V.

Any particular clustering solution can be presented in the form of an $N \times N$ cluster connectivity matrix $\mathcal{T}_{ij}$, defined by

$$\mathcal{T}_{ij} = \begin{cases} 1 & \text{points } i \text{ and } j \text{ belong to the same cluster} \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

In order to validate this solution, we construct an ensemble of m such matrices and make comparisons among them. This ensemble is created by constructing m resamples of the original data set. A resample is obtained by selecting at random a subset of size fN of the data points. We call $0 \leq f \leq 1$ the dilution factor.

We apply to every one of these subsets the same clustering procedure used on the full data set, using the same set of parameters V. This way we obtain for each resample μ, $\mu = 1, \ldots, m$, its own clustering results, summarized by the $fN \times fN$ matrix $\mathcal{T}^{(\mu)}$.

We define a figure of merit $\mathcal{M}(V)$ for the clustering procedure (and the choice of parameters) that we used. The figure of merit is based on comparing the connectivity matrices of the resamples, $\mathcal{T}^{(\mu)}$ ($\mu = 1, \ldots, m$), with the original matrix $\mathcal{T}$:

$$\mathcal{M}(V) = \langle\langle \delta_{\mathcal{T}_{ij}, \mathcal{T}^{(\mu)}_{ij}} \rangle\rangle_m. \tag{2.2}$$

The averaging implied by the notation $\langle\langle \cdot \rangle\rangle_m$ is twofold. First, for each resample μ, we average over all those pairs of points ij that were neighbors in the original sample and have both survived the resampling.[2] Second, this average value is averaged over all the m different resamples. Clearly, $0 \leq \mathcal{M} \leq 1$, with $\mathcal{M} = 1$ for perfect score.

The figure of merit $\mathcal{M}$ measures the extent to which the clustering assignments obtained from the resamples agree with those of the full sample. An important assumption we have made implicitly in this procedure is that the algorithm's parameters are "intensive," that is, their effect on the quality of the result is independent of the size of the data set. We can generalize our procedure in several ways to cases when this assumption does not hold (Levine, 2000).[3]

After calculating $\mathcal{M}(V)$, we have to decide whether we accept the clustering result, obtained using a particular value of the clustering parameters. For very low and very high values of $\mathcal{M}$, the decision may be easy, but for midrange values, we may need some additional information to guide our decision. In such a case, the best way to proceed is to change the values of the clustering parameters and go through the whole process once again.

Having done so for some set parameter choices V, we study the way $\mathcal{M}(V)$ varies as a function of V. Optimal sets of parameters V^* are identified by locating the maxima of this function. It should be noted, however, that some of these maxima are trivial and should not be used. Other maxima reflect different clustering solutions, which may all be considered valid.

[2] For various definition of neighbors, see Fukunaga (1990).

[3] For example, we can define our figure of merit on the basis of pairwise comparisons of our resamples and find an optimal set of parameters V_1^* in the way explained below. Next, we look for parameters V_2^* for which clustering of the full sample yields closest results to those obtained for the resamples (clustered at V_1^*).

For example, samples out of hierarchical distributions may have several meaningful solutions, corresponding to different levels of the hierarchy. Here we demonstrate the case of a single robust solution (examples with several solutions were given in Levine, 2000).

Let us denote by $A(N)$ the computational complexity of the clustering algorithm over a sample of size N with a given parameter set. The computational complexity of the resampling scheme just outlined can be estimated by $vmA(fN)$, where v is the number of different parameter sets among which one is searching for a best solution.

Our procedure can be summarized in the following algorithmic form:

Step 0. Choose values for the parameters V of the clustering algorithm.

Step 1. Perform clustering analysis of the full data set.

Step 2. Construct m subsets of the data set by randomly selecting fN of the N original data points.

Step 3. Perform clustering analysis for each subset.

Step 4. Based on the clustering results obtained in Steps 1 and 3, calculate $\mathcal{M}(V)$, as defined in equation 2.2.

Step 5. Vary the parameters V, and identify stable clusters as those for which a local maximum of $\mathcal{M}$ is observed.

3 Analysis of a One-Dimensional Model

To demonstrate the procedure outlined above, we consider a clustering problem that is simple enough to allow an approximate analytical calculation of the figure of merit $\mathcal{M}$ and its dependence on a parameter that controls the resolution. Consider a one-dimensional data set that consists of points x_i, $i = 1, \ldots, N$, selected from two identical but displaced uniform distributions, such as the one shown in Figure 3. The distributions are characterized by the mean distance between neighboring points, $d = 1/\lambda$, and the distance or gap between the two distributions, Δ. Distances between neighboring points within a cluster are distributed according to the Poisson distribution,

$$P(s) = \lambda e^{-\lambda s}\, ds. \tag{3.1}$$

The results of a clustering algorithm that reflects the underlying distribution from which the data were selected should identify two clusters in this data set.

Consider a simple nearest-neighbor clustering algorithm, which assigns two neighboring points to the same cluster if and only if the distance between the two is smaller than a threshold a. Clearly, $\alpha = \lambda a$ is the dimensionless parameter of the algorithm that controls the resolution of the clustering. For very small $\alpha \ll 1$, no two points belong to the same cluster, and the number of clusters equals N, the number of points; at the other extreme, $\alpha \gg 1$,

all pairs of neighbors are assigned to the same cluster, and hence all points reside in one cluster. Starting from $\alpha \gg 1$ and reducing it gradually, one generates a dendrogram. At an intermediate value of α, we may get any number of clusters between one and N. Hence, if we picked some particular value of α at which we obtained some clusters, we must face the dilemma of deciding whether these clusters are "natural" or the result of the fluctuations (i.e., noise) present in the data. In other words, we would like to validate the clustering solution obtained for the full data set for a given value of α. We do this using the resampling scheme described above.

A resample is generated by independently deciding for each data point whether it is kept in the resample (with probability f) or is discarded (with probability $1-f$). This procedure is repeated m times, yielding m resamples. All length scales of the original problem get rescaled by the resampling procedure by a factor of $1/f$; the mean distance between neighboring points in the resampled set is $d' = 1/\lambda f$, and the distance between the two uniform distributions is $\Delta' = \Delta/f$. Clustering is therefore performed with a rescaled threshold $a' = a/f$ on any resample; the resolution parameter keeps its original value, $\alpha' = a'/d' = \alpha$.

We first wish to get an approximate analytical expression for the figure of merit $\mathcal{M}(\alpha)$ described above. To do this, we consider the gaps between data points rather than the points themselves.

Let us denote by b the distance between the data point i (of the original sample) and its nearest left neighbor, $b = x_i - x_{i-1}$, with the two points on the same side of the gap Δ. We first assume that this edge is not broken by the clustering algorithm, $b < a$. Given a resample that includes point i, we define b' in the same fashion. The new, resampled left neighbor of i resides in the same cluster as i if $b' < a'$; the probability that this happens is given by (Levine, 2000)

$$P_1(\beta) = \sum_{m=1}^{\infty} \frac{f^2(1-f)^{m-1}}{(m-1)!}\gamma(m, \alpha/f - \beta), \tag{3.2}$$

where the dimensionless variable $\beta = \lambda b$ was introduced. Here $\gamma(n, z)$ is Euler's incomplete gamma function,

$$\gamma(n, z) = \int_0^z e^{-t} t^{n-1}\, dt, \tag{3.3}$$

except that in our convention, we take $\gamma(n, z < 0) = \gamma(n, 0)$.

Similarly, if points i and $i-1$ were not assigned to the same cluster in the original sample, then the probability that the same would happen in a resample is (Levine, 2000)

$$P_2(\beta) = \sum_{m=1}^{\infty} \frac{f^2(1-f)^{m-1}}{(m-1)!}\Gamma(m, \alpha/f - \beta), \tag{3.4}$$

where $\Gamma(n, z)$ is the other incomplete gamma function,

$$\Gamma(n, z) = \int_z^\infty e^{-t} t^{n-1}\, dt, \tag{3.5}$$

and we take $\Gamma(n, z < 0) = \Gamma(n, 0) = (n-1)!$, so $B_m = 1$ for $\alpha \leq f\beta$.

We now calculate the index $\mathcal{M}$ in an approximate manner by averaging P_1 and P_2 over all edges. This calculation matches the definition 2.2 of $\mathcal{M}$ in spirit. For pairs residing within a true cluster, the averaging is done by integrating over all possible values of b:

$$A(\alpha) = \int_0^\alpha e^{-\beta} P_1(\beta)\, d\beta + \int_\alpha^\infty e^{-\beta} P_2(\beta)\, d\beta. \tag{3.6}$$

For pairs that lie on different sides of the gap, we should compare a only with the size of the gap Δ,

$$B(\delta, \alpha) = \begin{cases} P_1(\delta) & \alpha \geq \delta \\ P_2(\delta) & \alpha < \delta, \end{cases} \tag{3.7}$$

where the dimensionless variable $\delta = \lambda\Delta$ was introduced.

Clearly, in the one-dimensional example, there are many fewer edges of the second kind than of the first. This, however, is not the case for data in higher dimensions, so we give equal weights to the two terms A and B,[4]

$$\mathcal{M}(\alpha) = \frac{1}{2}\left[A(\alpha) + B(\delta, \alpha)\right]. \tag{3.8}$$

We now plot $\mathcal{M}$ as a function of the resolution parameter α for both $f = 1/2$ and $f = 2/3$ (see Figure 2a), assuming the intercluster distance $\delta = 5$. A clear peak can be observed in both curves at $\alpha \simeq 2.5$ and $\alpha \simeq 3.3$, respectively. Similarly, for $\delta = 10$ clear peaks are identified at $\alpha \simeq 5$ and $\alpha \simeq 7$ (see Figure 2b). As we will see, these peaks correspond to the most stable clustering solution, which indeed recovers the original clusters. The trivial solutions of a single cluster (of all data points) and the opposite limit of N single-point clusters are also stable and appear as the maxima of $\mathcal{M}(\alpha)$ at $\alpha \ll 1$ and $\alpha \gg 1$, respectively.

In order to test how good our analytic approximate evaluation of $\mathcal{M}$ is, we clustered a one-dimensional data set of Figure 3 (top) and calculated the index $\mathcal{M}$ as defined in equation 2.2. The data set consists of $N = 300$ data points sampled from two uniform distributions of mean nearest-neighbor distance $1/\lambda = 1$ and shifted by $\Delta = 10$. The dendrogram obtained by varying α is shown in Figure 3 (bottom). It clearly exhibits two stable clusters,

[4] The ratio between the two terms is of the order $(d-1)/d$, where d is the dimensionality of the problem. For high-dimensional problems, this ratio is close to 1.

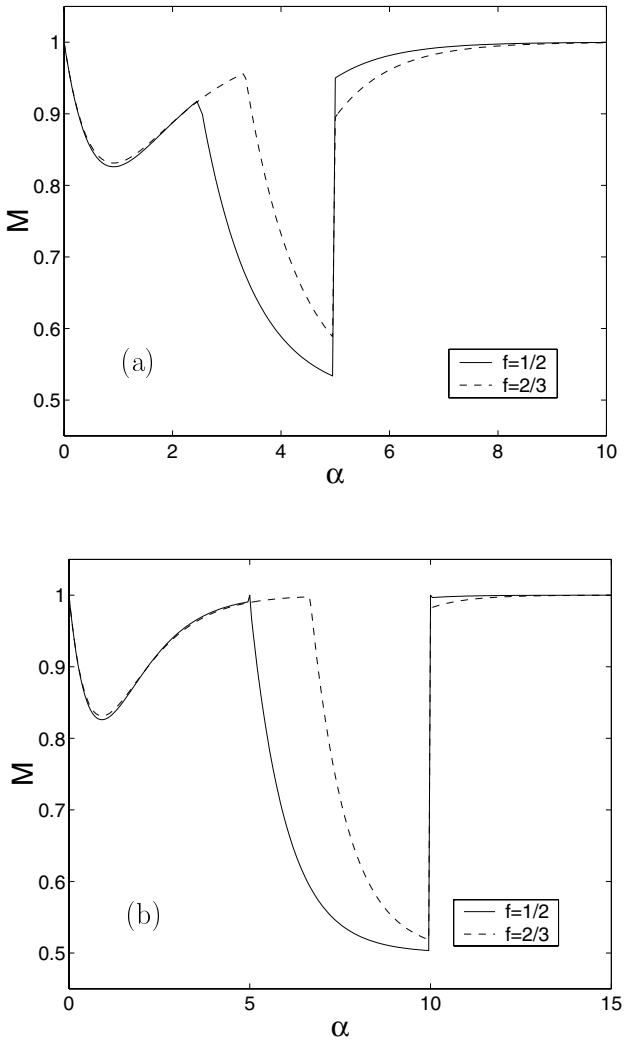

Figure 2: Mean behavior of $\mathcal{M}$ as a function of the geometric threshold, according to equation 3.8, for two clusters. The function is evaluated for the intercluster distances (a) $\Delta\lambda = 5$ and (b) $\Delta\lambda = 10$, with dilution parameters $f = 1/2$ and $f = 2/3$.

with stability indicated by the wide range of values of α over which the two clusters "survive." Next, we generated 100 resamples of size 150 (i.e., $f = 1/2$) and applied the geometrical clustering procedure described above to each resample.

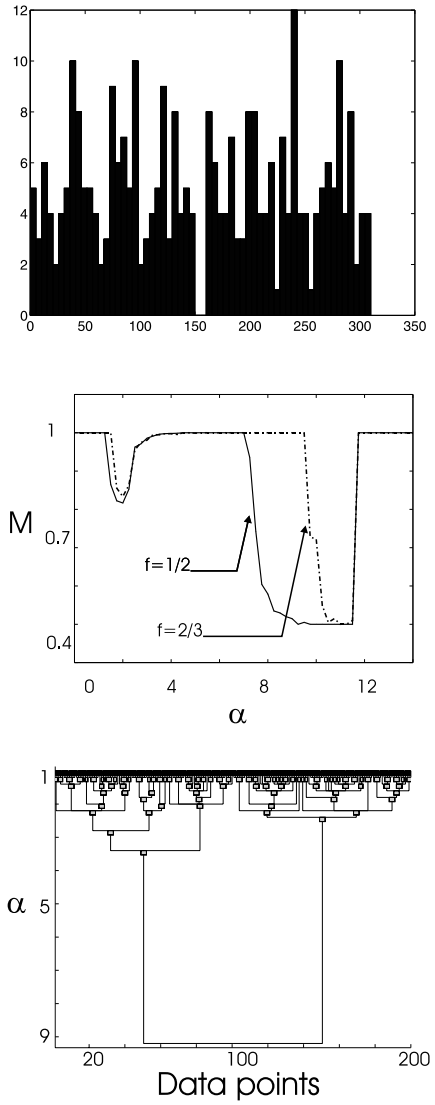

Figure 3: Resampling results for a one-dimensional data set. Three hundred points were chosen uniformly from two clusters, separated by $\Delta\lambda = 10$. The histogram of the data is given in the top panel. We performed 100 resamples of 150 points (i.e., $f = 1/2$) and 100 resamples of 200 points ($f = 2/3$), to calculate two versions of $\mathcal{M}$. In the middle panel, we plot $\mathcal{M}$ as a function of the resolution parameter α. The peak between $\alpha \approx 4$ and $\alpha \approx 7$ corresponds to the correct two-cluster solution, as can be seen from the dendrogram shown in the bottom panel.

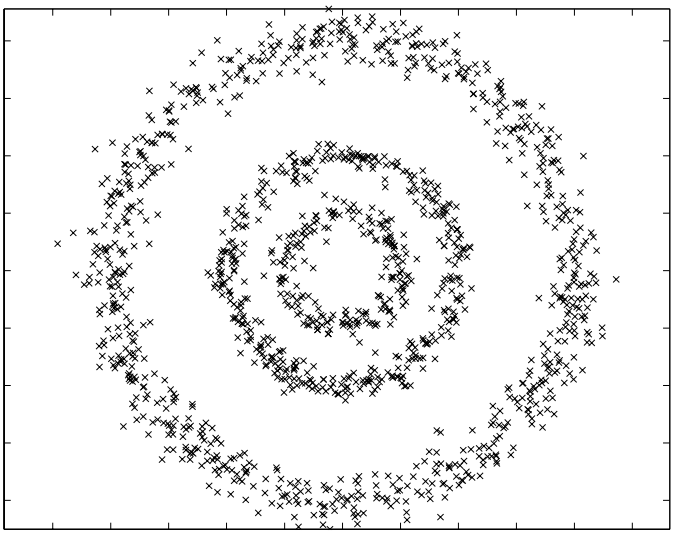

Figure 4: Three-ring problem. Fourteen hundred points are sampled from a three-component distribution (described in the text).

By averaging over the different resamples, the figure of merit $\mathcal{M}$ was calculated for different values of the dimensionless resolution parameter α, as shown in Figure 3 (middle). The peak between $\alpha \approx 4$ and $\alpha \approx 7$, corresponding to the most stable, "correct" clustering solution, is clearly identified. The whole procedure was also done with samples of size 200 ($f = 2/3$), giving similar results. In this case, as in Figure 2, the nontrivial peak extends to somewhat higher α-values when using larger f. In both cases, however, the location of the peak corresponds to the same, intuitively correct solution. The agreement between our approximate analytical curve of Figure 2b for $M(\alpha)$ and the numerically obtained exact curve of Figure 3 (middle) is excellent and most gratifying.

4 Applications

4.1 Two-Dimensional Toy Data. The analysis of the previous section predicts a typical behavior of $\mathcal{M}$ as a function of the parameters that control resolution; in particular, it suggests that one can identify a stable, "natural" partition as the one obtained at a local maximum of the function $\mathcal{M}$. This prediction was based on an approximate analytical treatment and backed up by numerical simulations of one-dimensional data. Here we demonstrate that this behavior is also observed for a toy problem, which consists of the two-dimensional data set shown in Figure 4. The angular coordinates of the data points are selected from a uniform distribution, $\theta \sim \mathrm{U}[0, 2\pi]$.

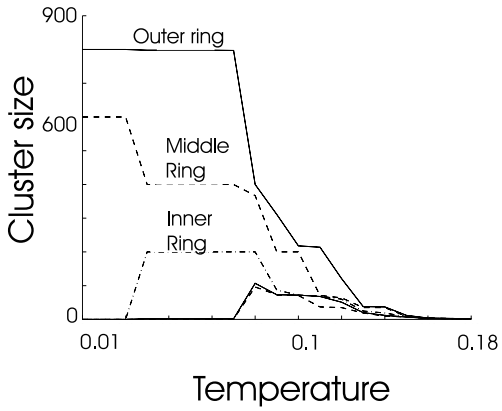

Figure 5: Clustering solution of the three-ring problem as a function of the resolution parameter (the temperature).

The radial coordinates are normal distributed, $r \sim \mathrm{N}[R, \sigma]$, around three different radii R. The outer "ring" ($R = 4.0$, $\sigma = 0.2$) consists of 800 points, and the inner "rings" ($R = 2.0, 1.0$, $\sigma = 0.1$) consist of 400 and 200 points, respectively.

The algorithm we chose to work with is the super-paramagnetic clustering (SPC) algorithm, recently introduced by Blatt et al. (1996; Domany, 1999). This algorithm provides a hierarchical clustering solution. A single parameter T, called "temperature," controls the resolution: higher temperatures correspond to higher resolutions. A variation of T generates a dendrogram. The outcome of the algorithm depends also on an additional parameter K, described in section 4.3.1. The data of Figure 4 were clustered with $K = 20$.

The results of the clustering procedure are presented in Figure 5. For $T < 0.02$, we get two clusters: the two inner rings constitute one cluster. A stable phase, in which the three rings are clearly identified as three clusters, appears in the temperature range $0.03 \leq T \leq 0.08$.

In order to identify the value of T that yields the "correct" solution, we generated and clustered 20 different resamples from this toy data set, with a dilution factor of $f = 2/3$. The resolution parameter (temperature) of each resample was rescaled so that the transition temperature at which the single large cluster breaks up agrees with the temperature of the same transition in the original sample.

The function $\mathcal{M}(T)$, plotted in Figure 6, exhibits precisely the expected behavior, with two trivial maxima and an additional one at $T = 0.05$. This value indeed corresponds to the "correct" solution of three clusters. Note that this maximum is not attained at the transition to the three-cluster solution ($T \simeq 0.03$) but at a slightly higher temperature ($T \simeq 0.05$). This shift

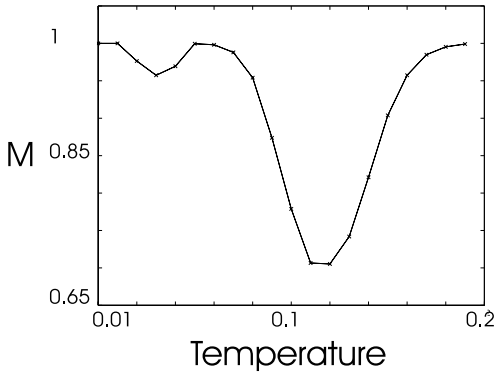

Figure 6: $\mathcal{M}$ as a function of the temperature for the three-ring problem.

reflects the fact that close to the transition, the two-cluster solution is still obtained for some resamples.

In order to compare our figure of merit with some known index, we applied several different clustering algorithms to the three-ring data set. For the K-means and DA algorithms, where the number of expected clusters must be provided, we searched for three clusters. A typical three-cluster solution generated by these two methods is shown in Figure 7; the three rings are not identified as clusters. Among the hierarchy of clustering solutions generated by SPC and average linkage algorithm (described in the next section), there appeared some that were very close to the intuitively correct one; we took these as the cluster assignement that is to be validated. We used our resampling scheme with 100 resamples and $f = 2/3$ for each of the methods. Although the solutions of SPC and average linkage yielded high $\mathcal{M}$ values (above 0.9), the results obtained by K-means and DA gave low values (below 0.6). We then calculated another index, R, defined by

$$R = \frac{1}{C} \sum_c \frac{S_c^{in}}{S_c^{out}}$$

$$S_c^{in} = \frac{1}{|c|^2} \sum_{i,j \in c} d_{ij}, \quad S_c^{out} = \min_{c'} \frac{1}{|c||c'|} \sum_{i \in c., j \in c'} d_{ij}, \tag{4.1}$$

where C is the number of clusters, d_{ij} is the distance between points i and j, and the summation is over all clusters (Jain & Dubes, 1988). This index measures the mean inner distance of the clusters relative to the distance from their nearest neighbor. Clearly this index looks for compact, well-separated clusters. It is therefore not surprising that of the clustering solutions proposed by the different methods, the one with the minimal value of R is that of Figure 7 obtained by the K-means algorithm ($R \simeq 0.56$, to be compared

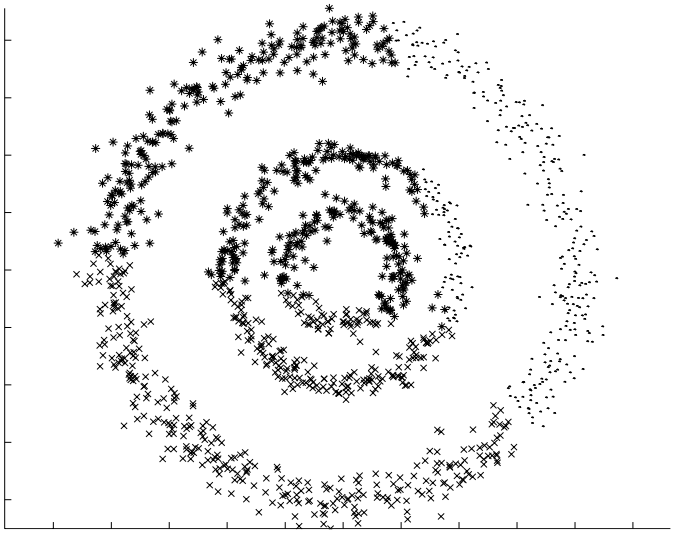

Figure 7: Clustering solution of the three-ring problem obtained by the K-means algorithm with $K = 3$.

with $R \simeq 0.87$ for the three-ring solution of SPC). We find that our figure of merit $\mathcal{M}$ assigns the highest score to the intuitively expected solution, whereas the R score is minimal for an "incorrect" solution.

4.2 A Single Cluster—Dealing with Cluster Tendency. A frequent problem of clustering methods is the so-called cluster tendency, that is, the tendency of the algorithm to partition any data, even when no natural clusters exist. In particular, agglomerative algorithms, which provide a hierarchy of clusters, always generate some hierarchy as a control parameter is varied. We expect this hierarchy to be very sensitive to noise, and thus unstable against resampling.

We tested this assumption for the data set of Figure 1a. The test was performed using two clustering methods: the SPC algorithm and the average-linkage clustering algorithm, an agglomerative hierarchical method. The average-linkage algorithm starts with N distinct clusters, one for each point, and forms the hierarchy by successively merging the closest pair of clusters, and redefining the distance between all other clusters and the new one. This step is repeated $N-1$ times until only a single element remains. The output of this hierarchical method is a dendrogram. (For more details, see Jain & Dubes, 1988.)

We performed our resampling scheme with $m = 20$ resamples and a dilution factor of $f = 2/3$, for different levels of resolution. The results are

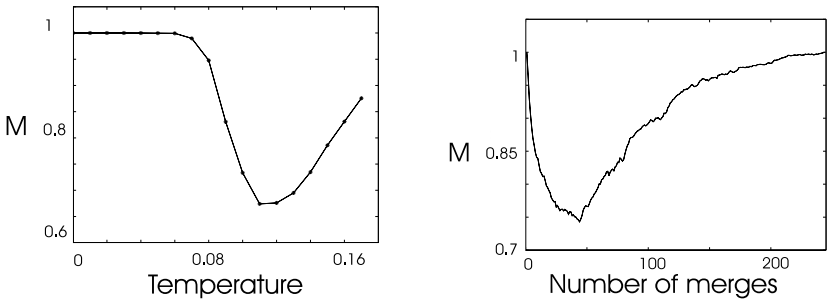

Figure 8: $\mathcal{M}$ as a function of the resolution parameter for the single-cluster problem. Clustering is performed using the SPC algorithm (left) and the average-linkage algorithm (right). The only natural "partitions" are a single cluster or N (single point) clusters.

shown in Figure 8. The only stable solutions identified by both algorithms are the trivial ones, of either a single cluster or N clusters. In this case, obviously the single-cluster solution is also the natural one.

4.3 Clustering DNA Microarray Data. We turn now to apply our procedure to real-life data. We present here validation of cluster analysis performed for DNA microarray data.

Gene-array technology provides a broad picture of the state of a cell by monitoring the expression levels of thousands of genes simultaneously. In a typical experiment, simultaneous expression levels of thousands of genes are viewed over a few tens of cell cultures at different conditions. (For details see Chee et al., 1996; Brown & Botstein, 1999.)

The experiment whose analysis is presented here is on colon cancer tissues (Alon et al., 1999). Expression levels of $n_g = 2000$ genes were measured for $n_t = 62$ different tissues, out of which 40 were tumor tissues and 22 were normal ones. Clustering analysis of such data has two aims:

- Searching for groups of tissues with similar gene expression profiles. Such groups may correspond to normal versus tumor tissues. For this analysis, the n_t tissues are considered as the data points, embedded in an n_g-dimensional space.
- Searching for groups of genes with correlated behavior. For this analysis, we view the genes as the data points, embedded in an n_t-dimensional space, and we hope to find groups of genes that are part of the same biological mechanism.

Following Alon et al., we normalize each data point (in both cases) such that the standard deviation of its components is one and its mean vanishes.

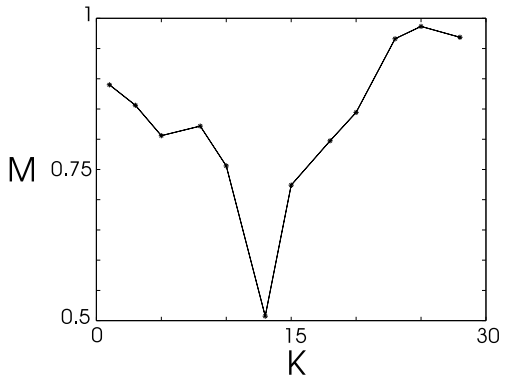

Figure 9: Figure of merit $\mathcal{M}$ as a function of K for the colon tissue data.

This way the Euclidean distance between two data points is trivially related to the Pearson correlation between the two.

4.3.1 Clustering Tissues. The main purpose of this analysis is to check whether one can distinguish between tumor and normal tissues on the basis of gene expression data. Since we know in what aspect of the data's structure we are interested, the working resolution in this problem is determined as the value at which the data first break into two (or more) large clusters (containing, say, more than 10 tissues).

There may be other parameters for the clustering algorithm, which should be determined. For example, the SPC algorithm has a single parameter K, which determines how many data points are considered as neighbors of a given point. The algorithm places edges or arcs that connect pairs of neighbors i, j, and assigns a weight J_{ij} to each edge, whose value decreases with the distance $|\vec{x}_i - \vec{x}_j|$ between the neighboring data points. Hence, the outcome of the clustering process may depend on K, the number of neighbors connected to each data point by an edge. We would like to use our resampling method to determine the optimal value for this parameter.

We clustered the tissues, using SPC, for several values of K. For each case, we identified the temperature at which the first split to large clusters occurred. For each case, we performed the same resampling scheme, with $m = 20$ resamples of size $\frac{2}{3}n_t$, and calculated the figure of merit $\mathcal{M}(K)$. The results obtained for several values of K are plotted in Figure 9. Very low and very high values of K give similarly stable solutions. In the low-K case, each point is practically isolated, and the data break immediately into microscopic clusters. The high-K case is just the opposite: each point is considered "close" to very many other points, and no macroscopic partition can be obtained.

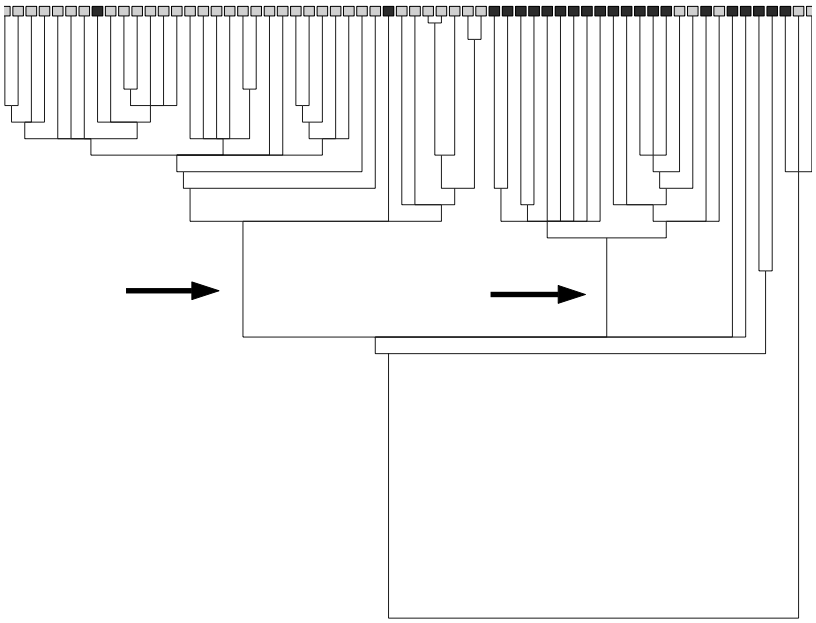

Figure 10: Clustering of the tissues, using $K = 8$. Leaves of the tree are colored according to the known classification to normal (dark) and tumor (light). The two large clusters marked by arrows recover the tumor versus normal classification.

At $K = 8$ we observe, however, another peak in $\mathcal{M}$. At this K, the clustering algorithm yields two large clusters, which correspond to the two "correct" clusters of normal and tumor tissues, as shown on Figure 10. Such solutions appear also for some higher values of K, but in these cases the clusters are not stable against resampling.

4.3.2 Clustering Genes. Cluster analysis of the genes is performed in order to identify groups of genes that act cooperatively. Having identified a cluster of genes, one may look for a biological mechanism that makes their expression correlated. Trying to answer this question, one may, for example, identify common promoters of these genes (Getz, Levine, Domany, & Zhang, 2000). One may also use one or more clusters of genes to reclassify the tissues (Getz, Levine, & Domany, 2000), looking for clinical manifestations associated with the expression levels of the selected groups of genes. Therefore, in this case, it is important to assess the reliability of each particular gene cluster separately.

The SPC clustering algorithm was used to cluster the genes. A resulting dendrogram is presented in Figure 11, in which each box represents a cluster

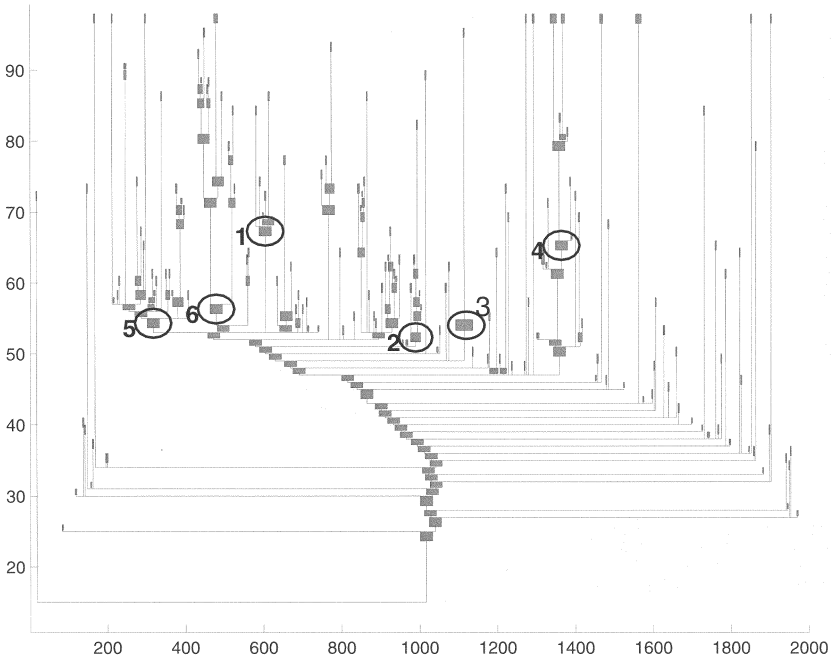

Figure 11: Dendrogram of genes. Clusters of size seven or larger are shown as boxes. Selected clusters are circled and numbered, as explained in the text.

of genes. Our resampling scheme has been applied to these data using $m = 25$ resamples of size 1200 ($f = 0.6$). This time, however, we calculated $\mathcal{M}(C)$ for each of the stable clusters C of the full original sample, one at a time. Since the structure of the dendrogram is sensitive to even slight changes in the data, we focused our attention only at the stable clusters emerging from clustering the full set of genes. For each stable cluster C, $\mathcal{M}$ was calculated at the temperature at which C was identified. We therefore get a stability measure for each cluster. Next, we focus our attention on clusters of the highest scores. First, we considered the top 20 clusters. If one of these clusters is a descendant of another one from the list of 20, we discard it. After this pruning, we were left with 6 clusters, which are circled and numbered in Figure 11 (the numbers are not related to the stability score).

We are now ready to interpret these stable clusters. The first three consist of known families of genes. Number 1 is a cluster of ribosomal proteins. The genes of cluster 2 are all cytochrome C genes, which are involved in energy transfer. Most of the genes of cluster 3 belong to the HLA-2 family, which are histocompatibility antigens.

Cluster 4 contains a variety of genes, some related to metabolism. When

trying to cluster the tissues based on these genes alone, we find a stable partition to two clusters, which is not consistent with the tumor versus normal labeling, but rather with a change in an experimental protocol (Getz, Levine, & Domany, 2000).

Clusters 5 and 6 also contain genes of various types. The genes of these clusters have the most typical behavior: All the genes of cluster 5 are highly expressed in the normal tissues but not in the tumor ones, and all genes of cluster 6 are the other way around.

To summarize, clustering stability score based on resampling enabled us to zero in on clusters with typical behavior, which may have biological meaning. Using resampling enabled us to select clusters without making any new assumption, a major advantage in exploratory research. The downside of this method, however, is its computational burden. We had to perform clustering analysis 20 times for a rather large data set. This would be the typical case for DNA microarray data.

5 Discussion

This work proposes a method to validate clustering analysis results, based on resampling. It is assumed that a cluster that is robust to resampling is less likely to be the result of a sample artifact or fluctuations.

The strength of this method is that it requires no additional assumptions. Specifically, no assumption is made about the structure of the data, the expected clusters, or the noise in the data. Only the available data are used.

We introduced a figure of merit, $\mathcal{M}$, which reflects the stability of the cluster partition against resampling. The typical behavior of this figure of merit as a function of the resolution parameter allows clear identification of natural resolution scales in the problem. Using this figure of merit, one can easily choose among different clustering solutions the one that is most robust. However, this figure of merit does not reflect the generalization performance of this clustering solution to new data.

The question of natural resolution levels is inherent in the clustering problem and thus emerges in any clustering scheme. The resampling method introduced here is general; it is applicable to any kind of data set and any clustering algorithm. Note, however, that if the data are so sparse that dilution can eliminate some of the underlying modes, our resampling scheme should not be used for cluster validation.

For a simple one-dimensional model, we derived an analytical expression for our figure of merit and its behavior as a function of the resolution parameter. Local maxima were identified for values of the parameter corresponding to stable clustering solutions. Such solutions can be either trivial (at very low and very high resolution) or nontrivial, revealing the genuine internal structure of the data.

Resampling is a viable method provided the original data set is large

enough, so that a typical resample still reflects the same underlying structure. If this is the case, our experience shows that a dilution factor of $f \simeq 2/3$ works well for both small and large data sets.

Acknowledgments

This research was partially supported by the Germany-Israel Science Foundation. We thank Ido Kanter and Gad Getz for discussions.

References

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc. Natl. Acad. Sci. USA, 96*, 6745–6750.

Bezdek, J. (1981). *Pattern recognition with fuzzy objective function algorithms*. New York: Plenum Press.

Bezdek, J., & Pal, N. (1995). Cluster validation with generalized Dunn's indices. *Proc. 1995 2nd NZ Int'l.* (pp. 190–193).

Blatt, M., Wiseman, S., & Domany, E. (1996). Super-paramagnetic clustering of data. *Physical Review Letter, 76*, 3251–3255.

Bock, H. (1985). On significance tests in cluster analysis. *J. Classification, 2*, 77–108.

Brown, P., & Botstein, D. (1999). Exploring the new world of the genome with DNA microarrays. *Nature Genetics, 21*, 33–37.

Chee, M., Yang, R., Hubbell, E., Berno, A., Huang, X. C., Stern, D., Winkler, J., Lockhart, D. J., Morris, M. S., & Fodor, S. P. A. (1996). Accessing genetic information with high-density DNA arrays. *Science, 274*, 610–614.

Cover, T., & Thomas, J. (1991). *Elements of information theory*. New York: Wiley-Interscience.

Cutler, A., & Windham, M. (1994). Information-based validity functionals for mixture analysis. In *Proc. 1st US/Japan Conf. on Frontiers in Statistical Modeling* (pp. 149–170).

Davies, D., & Bouldin, D. (1979). A cluster separation measure. *IEEE Trans. PAMI*, 224–227.

Domany, E. (1999). Super-paramagnetic clustering of data—the definitive solution of an ill-posed problem. *Physica A, 263*, 158.

Duda, R., & Hart, P. (1973). *Pattern classification and scene analysis*. New York: Wiley-Interscience.

Dunn, J. (1974). A fuzzy relative isodata process and its use in detecting compact well-separated clusters. *J. Cybernetics, 3*, 32–57.

Efron, B., & Tibshirani, R. (1993). *An introduction to the bootstrap*. London: Chapman & Hall.

Fukunaga, K. (1990). *Introduction to statistical pattern recognition*. San Diego: Academic Press.

Getz, G., Levine, E., & Domany, E. (2000). Coupled two-way clustering of DNA

microarray data. *Proc. Natl. Acad. Sci. USA, 97*, 12079.
Getz, G., Levine, E., Domany, E., & Zhang, M. (2000). Super-paramagnetic clustering of yeast gene expression profiles. *Physica A, 279*, 457.
Good, P. (1999). *Resampling methods*. New York: Springer-Verlag.
Jain, A., & Dubes, R. (1988). *Algorithms for clustering data*. Englewood Cliffs, NJ: Prentice Hall.
Jain, A., & Moreau, J. (1986). Bootstrap technique in cluster analysis. *Pattern Recognition, 20*, 547–569.
Levine, E. (2000). *Unsupervised estimation of cluster validity—Methods and applications*. Unpublished M.Sc. thesis, Weizmann Institute of Science.
Pal, N., & Bezdek, J. (1995). On cluster validity for the fuzzy c-means model. *IEEE Trans. Fuzzy Systems, 3*, 370–376.
Rose, K., Gurewitz, E., & Fox, G. (1990). Statistical mechanics and phase transitions in clustering. *Physical Review Letters, 65*, 945–948.
Smyth, P. (1996). Clustering using Monte Carlo cross-validation. In *KDD-96 Proceedings, Second International Conference on Knowledge Discovery and Data Mining* (pp. 126–133).
Windham, M. (1982). Cluster validity for the fuzzy c-means clustering algorithm. *IEEE Trans. PAMI, 4*, 357–363.

Received May 31, 2000; accepted January 4, 2001.