Clustering Objects on Subsets of Attributes

Jerome H. Friedman* Jacqueline J. Meulman†

March 14, 2004

Abstract

A new procedure is proposed for clustering attribute–value data. When used in conjunction with conventional distance based clustering algorithms this procedure encourages those algorithms to automatically detect subgroups of objects that preferentially cluster on *subsets* of the attribute variables rather than on all of them simultaneously. The relevant attribute subsets for each individual cluster can be different and partially (or completely) overlap with those of other clusters. Enhancements for increasing sensitivity for detecting especially low cardinality groups clustering on a small subset of variables are discussed. Applications in different domains, including gene expression arrays, are presented.

Keywords or phrases: distance based clustering, inverse exponential distance, clustering on variable subsets, targeted clustering, feature selection, mixtures of numeric and categorical variables, gene expression microarray data, genomics, bioinformatics

1 Introduction

The goal of cluster analysis is to partition a data set of N objects into subgroups such that those in each particular group are more similar to each other than to those of other groups. Defining an "encoder" function $c(i)$ that maps each object i to a particular group G_l $(1 \leq l \leq L)$

$$c(i) = l \Rightarrow i \in G_l, \tag{1}$$

one can formalize this goal as finding the "optimal" encoder $c^*(i)$ that minimizes a criterion $Q(c)$ that measures the degree to which the goal is not being met

$$c^* = \arg\min_c Q(c). \tag{2}$$

One such criterion is

$$Q(c) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} D_{ij}, \tag{3}$$

where D_{ij} is a defined distance or dissimilarity measure between every pair of objects (i, j), and N_l is the number of objects assigned to the lth group

$$N_l = \sum_{i=1}^{N} I(c(i) = l), \tag{4}$$

where the "indicator" function $I(\cdot) \in \{0, 1\}$ indicates truth of its argument, and where $\{W_l\}_1^L$ in (3) are cluster weights. Thus criterion (3) is a weighted average over the groups, of the

*Department of Statistics and Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94305 (jhf@stanford.edu)

†Data Theory Group, Leiden University, Leiden 2300 RB, The Netherlands (meulman@fsw.LeidenUniv.nl)

within group mean distance between pairs of objects assigned to the same group. The cluster weights $\{W_l\}_1^L$ are taken to be functions of the group sizes $\{N_l\}_1^L$ and can be used to regulate the distribution of groups sizes of the solution (2). (See Hubert, Arabie, & Meulman 2001, p.19 for a review of possible heterogeneity measures within a subset.) The usual choice $\{W_l = N_l^2\}_1^L$ gives the same influence to all object pairs in the criterion (3), encouraging equal sized solution clusters.

2 Attribute–value data

When each object i is characterized by a set of n measured attributes (variables),

$$\mathbf{x}_i = (x_{i1}, ..., x_{ik}, ..., x_{in}),$$

distances between pairs of objects D_{ij} as in (3) are based on their respective values (x_{ik}, x_{jk}) on each attribute k. A well-known example is the Gower (1971) coefficient of similarity. One defines a distance d_{ijk} between objects (i, j) separately on each attribute k, and then D_{ij} is taken to be a (weighted) average of the respective attribute distances

$$D_{ij} = \sum_{k=1}^{n} w_k d_{ijk} \tag{5}$$

with

$$\{w_k \geq 0\}_1^n \text{ and } \sum_{k=1}^{n} w_k = 1. \tag{6}$$

For example, the individual attribute distances can be taken as

$$d_{ijk} = \delta_{ijk}/s_k \tag{7}$$

where for numeric valued attributes

$$\delta_{ijk} = |\, x_{ik} - x_{jk} \,|, \tag{8}$$

or often its square, and for categorically valued (nominal) attributes

$$\delta_{ijk} = I(x_{ik} \neq x_{jk}). \tag{9}$$

There are numerous suggestions in the literature for distance measures on individual attributes other than (8) and (9). Particular choices reflect the goal of the cluster analysis. The approach presented in this paper applies to any such definitions. The denominator s_k (7) provides a scale for measuring "closeness" on each attribute. It is often taken to be

$$s_k = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \delta_{ijk} \tag{10}$$

or some other measure of spread or dispersion of the $\{x_{ik}\}_{i=1}^N$ values over all objects. For equal attribute weights $\{w_k = 1/n\}_1^n$, this gives the same influence to all of the attributes in defining the criterion (3) and thereby on the solution (2). Sometimes the weights in (5) are set to unequal values to further refine relative influence based on user domain knowledge or intuition, if it is suspected that particular attributes are more relevant than others to clustering the objects.

From (3) and (5) one can express the (equal weight) clustering criterion as

$$Q(c) = \sum_{l=1}^{L} W_l \left(\frac{1}{n} \sum_{k=1}^{n} S_{kl} \right) \tag{11}$$

where

$$S_{kl} = \frac{1}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} d_{ijk} \tag{12}$$

is a measure of the dispersion (scale) of the data values on the kth attribute for objects in the lth group, $\{x_{ik} \,|\, c(i) = l\}$. For example, if one uses $d_{ijk} = (x_{ik} - x_{jk})^2/s_k^2$ then $S_{kl} = 2 \cdot var\{x_{ik}/s_k \,|\, c(i) = l\}$. Thus, using (5) to define distance encourages (2)-(3) to seek clusters of objects that simultaneously have small dispersion on all or at least many of the attributes. That is, the objects within each solution subgroup are simultaneously close on a large number of the attributes.

3 Feature selection

Defining clusters in terms of simultaneous closeness on all attributes may sometimes be desirable, but often it is not. In data mining applications, the values of many attributes are often measured and it is unlikely that natural groupings will exist based on a large number of them. Usually, clustering, if it exists, occurs only within a relatively small unknown subset of the attributes. To the extent all of the attributes have equal influence, this type of clustering will be obscured and difficult to uncover.

The relative influence of each attribute x_k is regulated by its corresponding weight w_k in (5). Formally, feature selection seeks to find an optimal weighting $\mathbf{w} = \{w_k\}_1^n$ as part of the clustering problem by jointly minimizing the clustering criterion according to (3) and (5) with respect to the encoder c and weights $\mathbf{w}$. That is,

$$(c^*, \mathbf{w}^*) = \arg \min_{(c,w)} Q(c, \mathbf{w}) \tag{13}$$

where

$$Q(c, \mathbf{w}) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} D_{ij}[\mathbf{w}], \tag{14}$$

and $D_{ij}[\mathbf{w}]$ is given by (5), here emphasizing its dependence on the weights. The solution $\mathbf{w}^*$ has high weight values for those attributes that most exhibit clustering on the objects, and small values for those that do not participate in the clustering. The solution encoder c^* identifies the corresponding clusters. There is a vast literature on feature weighting/selection in clustering and classification; among others, see DeSarbo, Carroll, Clarck, & Green 1984, De Soete, DeSarbo, & Carroll 1985, De Soete 1986, 1988, Fowlkes, Gnanadesikan, & Kettenring 1988, Milligan 1989, Van Buuren & Heiser 1989, Gnanadesikan, Kettenring, & Tsao 1995, and Brusco & Cradit 2001.

4 Clustering on different subsets of attributes

Although feature selection is often helpful, it only seeks groups that all cluster on the same subset of attributes. Those are attributes with large solution weight values (13). However, individual clusters may represent groupings on different (possibly overlapping) attribute subsets, and it is of interest to discover such structure. With feature selection, clustering on *different* subsets of attributes will still be obscured and difficult to uncover.

One can generalize (14) to find clusters on separate attribute subsets by defining a separate attribute weighting $\mathbf{w}_l = \{w_{kl}\}_{k=1}^n$ for each individual group G_l, and jointly minimizing with respect to the encoder and all the separate weight sets associated with the respective groups. That is,

$$(c^*, \{\mathbf{w}_l^*\}_1^L) = \arg \min_{(c,\{\mathbf{w}_l\}_1^L)} Q(c, \{\mathbf{w}_l\}_1^L), \tag{15}$$

where

$$Q(c, \{\mathbf{w}_l\}_1^L) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} D_{ij}[\mathbf{w}_l], \tag{16}$$

and

$$D_{ij}[\mathbf{w}_l] = \sum_{k=1}^{n} w_{kl} d_{ijk}. \tag{17}$$

As before (6), the attribute weights satisfy

$$\{w_{kl} \geq 0\}_1^n \text{ and } \sum_{k=1}^{n} w_{kl} = 1, \quad 1 \leq l \leq L. \tag{18}$$

For any given encoder c, the solution to (15)-(16) for the corresponding attribute weights is $w_{kl}^* = I(k = k_l^*)$ where $k_l^* = \arg\min_{1 \leq k \leq n} S_{kl}$, with S_{kl} given by (12). That is, the solution will put maximal (unit) weight on that attribute with smallest dispersion within each group G_l, and zero weight on all other attributes regardless of their respective dispersions within the group. Therefore, minimizing criterion (16) will produce solution groups

that tend to cluster only on a *single* attribute. This type of clustering can be detected by simple inspection of the marginal data distributions on each attribute separately. Our goal is finding groups of objects that simultaneously cluster on subsets of attributes, where each subset contains more than one attribute.

This goal can be accomplished by modifying the criterion (16) with an incentive (negative penalty) for solutions involving more attributes. One such incentive is the negative entropy of the weight distribution for each group

$$e(\mathbf{w}_l) = \sum_{k=1}^{n} w_{kl} \log w_{kl}. \tag{19}$$

This function achieves its minimum value for equal weights and is correspondingly larger as the weights become more unequal. Incorporating (19), the modified criterion becomes

$$Q(c, \{\mathbf{w}_l\}_1^L) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} D_{ij}^{(\lambda)}[\mathbf{w}_l], \tag{20}$$

with

$$D_{ij}^{(\lambda)}[\mathbf{w}_l] = \sum_{k=1}^{n} (w_{kl}\, d_{ijk} + \lambda\, w_{kl} \log w_{kl}) + \lambda \log n. \tag{21}$$

(The last term simply provides a translation so that $\min_{\mathbf{w}_l} D_{ij}^{(\lambda)}[\mathbf{w}_l] = 0$ whenever $\{d_{ijk} = 0\}_{k=1}^n$.) The quantity $\lambda \geq 0$ controls the strength of the incentive for clustering on more attributes. It is a meta–parameter of the procedure and provides control over the type of clustering sought. Increasing/decreasing its value will encourage clusters on more/less attributes.

For a given encoder c, the solution to (15), minimizing (20)-(21) for the corresponding optimizing weight values is

$$w_{kl} = \exp(-S_{kl}/\lambda) \Big/ \sum_{k'=1}^{n} \exp(-S_{k'l}/\lambda), \tag{22}$$

with S_{kl} given by (12). This solution puts increased weight on attributes with smaller dispersion within each group G_l, where degree of this increase is controlled by the value of λ. Setting $\lambda = 0$ places all weight on the attribute k with smallest S_{kl}, whereas $\lambda = \infty$ forces all attributes to be given equal weight for each group G_l.

Since all individual attribute distances (7) are normalized as in (10), the quantity S_{kl} will tend to be near unity for attributes that do not contribute to the clustering of group G_l, and smaller for those that do. In this sense the value chosen for λ defines the meaning of "clustering" on an attribute. A group of objects $G_l = \{i \mid c(i) = l\}$ is said to "cluster" on attribute k, if S_{kl} for group G_l is smaller than the value of λ. Considerations governing the choice for its value are discussed in Section 6.1 below.

For a given encoder c, one can minimize (20)-(21) with respect to all the weights $\{\mathbf{w}_l\}_1^L$ (22), thereby producing a criterion $Q(c)$ that depends only on the encoder. The result is

$$Q(c) = \sum_{l=1}^{L} W_l \cdot \left[-\lambda \log \left(\frac{1}{n} \sum_{k=1}^{n} \exp(-S_{kl}/\lambda) \right) \right], \tag{23}$$

where the optimal encoder is given by (2). The bracketed quantity in (23) is proportional to a generalized (Orlicz) mean

$$f^{-1} \left[\frac{1}{n} \sum_{k=1}^{n} f(S_{kl}) \right] \tag{24}$$

of $\{S_{kl}\}_{k=1}^n$, where here

$$f(z) = 1/\exp(z/\lambda) \tag{25}$$

is the inverse exponential function with scale parameter λ. This criterion (23) can be contrasted with that for ordinary clustering (11). Clustering based on distances using (5) with equal (or other prespecified) attribute weights minimizes the *arithmetic* mean of the attribute dispersions within each cluster; separate optimal attribute weighting within each cluster of objects minimizes the *inverse exponential* mean (24)-(25).

5 Search strategy

Defining the clustering solution as the minimum of some criterion does not fully solve the problem. One needs a method for finding the minimizing encoder c^* that identifies the solution clusters. This is a combinatorial optimization problem (among others, see Hansen & Jaumard 1997, Hubert, Arabie, & Meulman, 2001, Van Os, 2001) for which a complete enumeration search over all possible encoders is computationally impractical for large problems. For these one must employ less than thorough heuristic search strategies.

For ordinary clustering based on (3) or similar criteria, a large number of heuristic search strategies have been proposed. These are known as distance based "clustering algorithms" (for example, see Hartigan 1975, Späth 1980, Jain & Dubes 1988, Kaufman & Rousseeuw 1990, Arabie, Hubert, & De Soete 1996, Mirkin 1996, Gordon 1999). For attribute–value data (Section 2), clustering algorithms equivalently attempt to minimize (11) by using (5) with equal (or prespecified) weights to define the distances D_{ij} between object pairs.

The criterion (23) is a more complicated highly non convex function of the $\{S_{kl}\}$. The approach used here is to apply an alternating optimization strategy based on (20). One starts with an initial guess for the weight values, for example all values equal $\{w_{kl} = 1/n\}$. The criterion (20) is then minimized with respect to the encoder given those weight values. Given that encoder, (20) is minimized with respect to the weights, producing a new set of values for $\{\mathbf{w}_l\}_1^L$. These are then used to solve for a new encoder, and so on. This iterative procedure is continued until a (local) minimum is reached.

From (21) the criterion (20) can be expressed as

$$Q(c, \{\mathbf{w}_l\}_1^L) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} D_{ij}[\mathbf{w}_l] + \lambda \sum_{l=1}^{L} W_l \sum_{k=1}^{n} w_{kl} \log w_{kl}, \tag{26}$$

where $D_{ij}[\mathbf{w}_l]$ is given by (17).

For a given encoder $c(\cdot)$, the minimizing solution for the weights is given by (22). Given a set of weight values $\mathbf{W} = \{\mathbf{w}_l\}_1^L \in R^{n\times L}$, the solution encoder $c^*(\cdot\,|\mathbf{W})$ minimizes

$$Q(c\,|\mathbf{W}) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i\,|\mathbf{W})=l} \sum_{c(j\,|\mathbf{W})=l} D_{ij}[\mathbf{w}_l]. \tag{27}$$

The form of this criterion is similar to that of (3) where distance between objects assigned to the same group, $c(i\,|\mathbf{W}) = c(j\,|\mathbf{W}) = l$, is given by $D_{ij}[\mathbf{w}_l]$. However, conventional clustering algorithms cannot be directly used to attempt to minimize (27) since they require distances to be defined between all object pairs, not just those assigned to the same group. The strategy employed here is to define a distance $D_{ij}[\mathbf{W}]$ between *all* object pairs that when used with standard clustering algorithms produces an encoder that approximates the solution $c^*(\cdot\,|\mathbf{W})$ minimizing (27).

The starting point for deriving such a distance measure is the assumption that $c^*(\cdot\,|\mathbf{W})$ has the property

$$\frac{1}{N_l^2} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=l} D_{ij}[\mathbf{w}_l] < \frac{1}{N_l N_m} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=m} D_{ij}[\mathbf{w}_l], \quad m \neq l, \tag{28}$$

for all solution groups $G_l = \{i\,|\,c^*(i\,|\mathbf{W}) = l\}$. That is, the average distance between pairs of objects *within* the same group G_l, based on the weights for that group $\mathbf{w}_l$, is smaller than the corresponding average distance *between* groups based on $\mathbf{w}_l$. If this were not the case, the value of (27) could be further reduced by merging G_l with all groups G_m $(m \neq l)$ for which (28) was violated. Furthermore from (28) one has

$$\frac{1}{N_l^2} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=l} D_{ij}[\mathbf{w}_l] < \frac{1}{N_l N_m} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=m} \max(D_{ij}[\mathbf{w}_l], D_{ij}[\mathbf{w}_m]) \tag{29}$$

for $m \neq l$. Therefore, defining

$$D_{ij}^{(1)}[\mathbf{W}] = \max(D_{ij}[\mathbf{w}_{c(i\,|\mathbf{W})}], D_{ij}[\mathbf{w}_{c(j\,|\mathbf{W})}]) \tag{30}$$

one has

$$\frac{1}{N_l^2} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=l} D_{ij}[\mathbf{w}_l] = \frac{1}{N_l^2} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=l} D_{ij}^{(1)}[\mathbf{W}], \tag{31}$$

and from (29)

$$\frac{1}{N_l^2} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=l} D_{ij}^{(1)}[\mathbf{W}] < \frac{1}{N_l N_m} \sum_{c^*(i\,|\mathbf{W})=l} \sum_{c^*(j\,|\mathbf{W})=m} D_{ij}^{(1)}[\mathbf{W}], \quad m \neq l. \tag{32}$$

That is, the solution encoder $c^*(\cdot\,|\mathbf{W})$ minimizing (27) has the property that the average within group distance, using (30), is smaller the the corresponding between group average. The solutions produced by standard clustering algorithms also attempt to achieve this goal. Therefore, applying a standard clustering algorithm based on $D_{ij}^{(1)}[\mathbf{W}]$ (30) will attempt to produce a solution minimizing (27).

The distance define by (30) is not the only one that satisfies properties (31) (32). Any "majorizing" distance that is equal to $D_{ij}^{(1)}[\mathbf{W}]$ when $c(i\,|\mathbf{W}) = c(j\,|\mathbf{W})$ and is larger otherwise will share these properties. An example is

$$D_{ij}^{(2)}[\mathbf{W}] = \sum_{k=1}^{n} \max(w_{k,c(i\,|\mathbf{W})}, w_{k,c(j\,|\mathbf{W})})\, d_{ijk}. \tag{33}$$

Any such distance could be used to produce a surrogate criterion for (27) in the form of (3) to be minimized by conventional clustering algorithms. Specific choice will depend on performance

in the context of a particular clustering algorithm. This situation is common in optimization problems using heuristic search strategies, where one often chooses to optimize a surrogate criterion with the same solution as the desired one. The choice of a surrogate is based solely on performance in the context of the chosen search strategy. Empirical evidence so far suggests that both (30) and (33) yield similar results using common clustering algorithms, with (33) sometimes providing superior performance.

In summary, an alternating optimization algorithm attempting to minimize (23) would initialize all weight values to $\mathbf{W} = \{w_{kl} = 1/n\}$. A solution encoder $c^*(\cdot\,|\mathbf{W})$ is obtained by applying a clustering algorithm using either (30) or (33) to define interpoint distances. New weight values $\mathbf{W}$ are computed based on $c^*(\cdot\,|\mathbf{W})$ using (12) (22). These weight values define new inter object distances for the clustering algorithm. These steps are iterated until the solution stabilizes.

6 Weighted inverse exponential distance

The alternating optimization strategy outlined in the previous section is unlikely to produce satisfactory results if applied straightforwardly. The highly non convex nature of (23) induces a very large number of distinctly suboptimal local solutions. If the initial weight values $\mathbf{W} = \{1/n\}$ are far from their (global) minimizing values, it is likely that the alternating strategy will converge to one of these suboptimal local solutions. This will especially be the case when there is clustering on small subsets of the attributes. In order to be successful, it is necessary either to find good initial weight values close to the solution values, or to use an alternative surrogate criterion for which the weight values $\mathbf{W} = \{1/n\}$ provide a good starting point. Since it is usually difficult to assign good starting values without knowing the ultimate solution, the latter strategy is pursued here.

For any set of weights $\mathbf{w} = \{w_k\}_1^n$ consider the interpoint distance measure

$$\begin{aligned} D_{ij}^{(\eta)}[\mathbf{w}] &= \min_{\{t_k\}_1^n} \sum_{k=1}^{n} t_k\, d_{ijk} + \eta\, t_k\, \log \frac{t_k}{w_k}, \quad \sum_{k=1}^{n} t_k = 1, \\ &= -\eta \log \sum_{k=1}^{n} w_k e^{-d_{ijk}/\eta}. \end{aligned} \tag{34}$$

This is a distance between objects (i, j) based on a weighted inverse exponential mean (24) (25) of $\{d_{ijk}\}_{k=1}^n$ with scale parameter η.

As η becomes large , $D_{ij}^{(\eta)}[\mathbf{w}]$ approaches the ordinary distance (5),

$$\lim_{\eta\to\infty} D_{ij}^{(\eta)}[\mathbf{w}] = \sum_{k=1}^{n} w_k\, d_{ijk}. \tag{35}$$

Therefore, as the limit is approached this distance definition (34) can be used on the right hand side of (30) or (33) to produce equivalent surrogate criteria for (27).

For finite values of η alternate surrogate criteria are defined. These alternatives need not lead to equivalent surrogates for (27) since they will not necessarily satisfy properties (31) (32). However, setting the value of η in (34) to be the same as that used for λ in (23) produces a criterion quite similar to (23) when all weight values are taken to be equal $\mathbf{W} = \{1/n\}$. This can be seen by first using (12) to express (23) as

$$Q(c) = -\lambda \sum_{l=1}^{L} W_l \log \frac{1}{n} \sum_{k=1}^{n} \left(\prod_{\substack{c(i)=l \\ c(j)=l}} e^{-d_{ijk}/\lambda} \right)^{\frac{1}{N_l^2}}. \tag{36}$$

Setting $\eta = \lambda$ and then substituting (34) into (30) or (33) with all weight values equal to $1/n$ produces the surrogate criterion

$$\tilde{Q}(c) = -\lambda \sum_{l=1}^{L} W_l \log \left(\prod_{\substack{c(i)=l \\ c(j)=l}} \frac{1}{n} \sum_{k=1}^{n} e^{-d_{ijk}/\lambda} \right)^{\frac{1}{N_l^2}}. \tag{37}$$

Each term in both (36) and (37) contains the logarithm of a measure of central tendency of $\{e^{-d_{ijk}/\lambda}\}$, for all $c(i) = c(j) = l$ and $1 \leq k \leq n$. For $Q(c)$ this measure is the arithmetic mean over k of the geometric mean over (i, j). For $\tilde{Q}(c)$ it is the geometric mean over (i, j) of the arithmetic mean over k. Both of these criteria are similar in that they are most strongly influenced by the d_{ijk} that have small values compared to λ, and correspondingly less influenced by those with larger values. By contrast, directly using (30) or (33) based on (17) using equal weight values $\mathbf{W} = \{1/n\}$ produces the criterion

$$\bar{Q}(c) = \sum_{l=1}^{L} \frac{W_l}{N_l^2} \sum_{c(i)=l} \sum_{c(j)=l} \frac{1}{n} \sum_{k=1}^{n} d_{ijk}. \tag{38}$$

Each term in (38) contains a measure of central tendency of $\{d_{ijk}\}$, for all $c(i) = c(j) = l$ and $1 \leq k \leq n$, based on the arithmetic mean. This criterion is independent of λ and most strongly influenced by the larger valued d_{ijk}. Therefore, to the extent that the respective geometric and arithmetic means of $\{e^{-d_{ijk}/\lambda}\}$ appearing in (36) and (37) are not too different, solutions minimizing (37) would likely be much closer to those minimizing (23) (36) than solutions produced by minimizing (38). (Note that $0 < e^{-d_{ijk}/\lambda} \leq 1$.) Empirical evidence suggests that this is indeed the case.

Since $Q(c)$ and $\tilde{Q}(c)$ are not identical, applying a clustering algorithm based on (30) or (33), substituting (34) in place of (17) (with $\eta = \lambda$ and equal weights $\mathbf{W} = \{1/n\}$) does not produce the solution minimizing (23). It only provides a potentially good starting point for the iterative algorithm described in Section 5. From (35), as $\eta \to \infty$ this substitution produces the distance measure used by that algorithm. This suggests a homotopy optimization strategy in which (34) replaces (17) in (30) or (33), with η being the homotopy parameter. Its value is initialized to that of λ, and then gradually increased as iterations proceed. This smoothly transforms the criterion being minimized from (37), to (27) based on (30) or (33), as the weight values progress from $\mathbf{W} = \{1/n\}$ to their minimizing values. Such a strategy leads to the following algorithm for clustering objects on subsets of attributes (COSA):

Algorithm COSA1

1 Initialize: $\mathbf{W} = \{1/n\}$; $\eta = \lambda$
2 Loop {
3 Compute distances $D_{ij}[\mathbf{W}]$ (30) (33) (34)
4 $c \leftarrow$ clustering algorithm ($\{D_{ij}[\mathbf{W}]\}$)
5 Compute weights $\mathbf{W} = \{\mathbf{w}_l\}_1^L$ (12) (22)
6 $\eta = \eta + \alpha \cdot \lambda$
7 } Until $\mathbf{W}$ stabilizes
8 Output: $c^* = c$

For a given value of λ, the value of α (line 6) controls the rate of increase in the value of the homotopy parameter η. There is as yet no theory to suggest appropriate values of α in particular applications. Setting $\alpha = \infty$ causes this algorithm to compute the solution minimizing (37) at the first iteration, and then immediately to switch to the algorithm described in Section 5, based on ordinary distance, (17), (30) or (33), starting at that solution. Smaller values of α cause a more gradual evolution from weighted inverse exponential distance (34) to ordinary distance

as the weight values in turn evolve. Empirical evidence suggests that if the clustered groups tend to concentrate on small subsets of the attributes the value of α should be taken to be fairly small ($\alpha \lesssim 0.1$), causing a slow evolution. Otherwise, the weighted inverse exponential distance approaches ordinary distance too rapidly, thereby causing the algorithm to converge to an inferior local minimum in spite of its potentially good starting point.

In order for inverse exponential distance (34) to closely approximate ordinary distance (35) the value of the homotopy parameter η must become large compared to typical values of the interpoint distances d_{ijk} on each attribute k. As a consequence of their normalization (7) (10) these attribute interpoint distances have expected values of unity with typical values in the range $0 \lesssim d_{ijk} \lesssim 2$. For example, with normally distributed attribute values $\{x_{ik}\}$ and equal weights $\{w_k = 1/n\}_1^n$, the correlation of the distances (34) for $\eta = 1$ with those produced by $\eta = \infty$ (35) is already 0.97, and for $\eta = 2$ it is 0.99. This suggests that the transition of (34) to (35) is achieved when η reaches values in this range.

The algorithm however usually converges to equivalent solutions for much smaller values of η. This is caused by the weighting of the respective attributes within each clustered group after the first iteration (line 5). Attributes k with typically large interpoint distances ($d_{ijk} >> \lambda$) receive small weights through (12) (22) compared to those characterized by small interpoint distances ($d_{ijk} \lesssim \lambda$). Thus, both distance measures (34) (35) are primarily influenced by those attributes k for which $d_{ijk} \lesssim \lambda$. This mechanism causes the transition of (34) to (35) to occur for much smaller values of η, typically $\eta \simeq \lambda$, when there is preferential clustering on attribute subsets. In fact, for all the applications presented in Section 12 below, using $0 \leq \alpha \leq 0.25$ caused the algorithm to converge to equivalent solutions, with some being invariant over a much broader range.

In applications where clustering tends to occur on relatively large numbers of attributes, larger values of α will tend to produce better results. However, it is in precisely these settings that the usual clustering algorithms (3) (5) based on unweighted distances $\{w_k = 1/n\}_1^n$ perform well, and there is little advantage associated with the COSA strategy.

6.1 Scale parameter

The primary tuning parameter of the COSA procedure is the scale parameter λ (22). The goal is to identify groups of objects $G_l = \{i \,|\, c(i) = l\}$ (clusters) such that on subsets of the attributes k, the characteristic interpoint distances d_{ijk} within each are relatively small $\{d_{ijk} << 1 \,|\, c(i) = c(j) = l\}$. From (12) (22) the value of λ defines the characteristic scale of these "small" interpoint distances to which the procedure will have sensitivity. For large values $\lambda \gtrsim 1$ ($\eta \geq \lambda$) both (34) and (35) reduce to ordinary distance with equal weights $\{w_{kl} = 1/n\}_1^n$ on all attributes within each cluster, so that COSA approximates ordinary clustering based on (3) (5). Thus if the goal is to uncover preferential clustering on subsets of attributes the value of λ should be taken to be small ($\lambda << 1$).

As the value of λ is reduced however, fewer objects within each group G_l have influence on the estimated weights through (12) (22), thereby increasing the variance of these estimates and reducing the power of the procedure. Thus, λ can be regarded as "smoothing" parameter controlling a kind of bias–variance trade–off in analogy with more general density estimation procedures. Values that are too large give rise to over–smoothing reducing sensitivity to narrow clustering on small subsets of attributes. Values of λ that are too small (under–smoothing) increase variance that also reduces power to uncover the overall clustering structure. Ideally, the value of λ should be set to the characteristic scale of the small distances d_{ijk} on those attributes k upon which each of the groups G_l preferentially cluster. This is of course unknown. Variance considerations suggest somewhat larger values for smaller sample sizes.

Since an optimal value of λ is situation dependent and there is as yet no theory to suggest good values, the only recourse is to experiment with several values and examine the results. Empirical evidence so far suggests that in the presence of sharp clustering on small subsets of attributes the procedure is usually not highly sensitive to values in the range $0.1 \leq \lambda \leq 0.4$.

However, in the presence of more subtle structure the results can be fairly sensitive to a choice for its value.

7 Hierarchical Clustering

The COSA1 algorithm of the preceding section uses a conventional iterative clustering method as a primitive (line 4). It can be viewed as a "wrapper" placed around a chosen clustering algorithm extending that algorithm to clustering on subsets of attributes. As with most conventional iterative clustering methods, the number of clusters sought L must be specified.

A very popular class of clustering techniques, especially with gene expression microarray data, are hierarchical methods. These do not require prespecification of the number of clusters. Instead, they arrange potential clusters in a hierarchy displayed as a binary tree ("dendrogram"). The user can then visualize this representation to assess the degree of clustering present in the data, and manually choose a particular partition of the objects into groups. Using COSA1 as a wrapper around such a manually driven procedure is cumbersome at best. For hierarchical clustering, one needs a version of the algorithm that provides inter object distances $\{D_{ij}\}$ encouraging clustering on subsets of attributes, without requiring the specification of a particular iterative clustering algorithm or the number of groups L.

The key ingredient to producing such a version is based on the definition of clustering: pairs of objects (i, j) within the same solution clustered group $c^*(i) = c^*(j)$, using a particular distance definition D_{ij}, will tend to have relatively small values of D_{ij}. This is the goal driving all clustering methods. Let $KNN(i)$ be K closest objects to i based on D_{ij},

$$KNN(i) = \{j \,|\, D_{ij} \leq d_{i(K)}\} \tag{39}$$

where $d_{i(K)}$ is the Kth order statistic of $\{D_{ij}\}_{j=1}^{N}$ sorted in ascending values. Then among those objects $j \in KNN(i)$ there will be an over representation of objects for which $c^*(i) = c^*(j)$. That is,

$$\frac{1}{K} \sum_{j \in KNN(i)} I\left[c^*(j) = c^*(i)\right] > \frac{1}{N} \sum_{j=1}^{N} I\left[c^*(j) = c^*(i)\right]. \tag{40}$$

The more pronounced the clustering, the stronger this inequality becomes. Therefore, to the extent (40) holds, statistics computed on $KNN(i)$ will reflect those computed on $\{j \,|\, c^*(j) = c^*(i)\}$. In particular, for the scale measure (12) this implies

$$S_{k,c^*(i)} \backsimeq \frac{1}{K^2} \sum_{j \in KNN(i)} \sum_{j' \in KNN(i)} d_{jj'k}. \tag{41}$$

This represents a measurement of scale of the attribute x_k for objects $\{j \,|\, j \in KNN(i)\}$. Furthermore, in the interest of reduced computation (41) can in turn be approximated by

$$S_{ki} = \frac{1}{K} \sum_{j \in KNN(i)} d_{ijk}. \tag{42}$$

Under these assumptions one can modify the COSA1 algorithm by replacing the clustering algorithm (line 4) by a procedure that computes $\{KNN(i)\}_1^N$, and replacing $\mathbf{w}_{c(i\,|\,\mathbf{W})} \leftarrow \mathbf{w}_i = \{w_{ki}\}_{k=1}^{n}$ in (30) (33) (34) for computing the distances (line 3), with

$$w_{ki} = \exp(-S_{ki}/\lambda) \Big/ \sum_{k'=1}^{n} \exp(-S_{k'i}/\lambda) \tag{43}$$

for calculating the weights (line 5). With this substitution, the matrix of weights $\mathbf{W}$ becomes an $n \times N$ matrix with entries w_{ki}. These changes produce the following algorithm:

Algorithm COSA2

1 Initialize: $\mathbf{W} = \{1/n\}$; $\eta = \lambda$
2 Loop {
3 Compute distances $D_{ij}[\mathbf{W}]$ (30) (33) (34)
4 Compute $\{KNN(i)\}_1^N$ (39)
5 Compute weights $\mathbf{W} = \{w_{ki}\}$ (42) (43)
6 $\eta = \eta + \alpha \cdot \lambda$
7 } Until $\mathbf{W}$ stabilizes
8 Output: $\{D_{ij} = D_{ij}[\mathbf{W}]\}$

The purpose of this algorithm is to obtain a good set of weight values $\mathbf{W} \in \mathbf{R}^{n \times N}$ for calculating interpoint distances $\{D_{ij}\}$ (line 8) by approximately minimizing the criterion

$$Q(\mathbf{W}) = \sum_{i=1}^{N} \left[\frac{1}{K} \sum_{j \in KNN(i)} D_{ij}[\mathbf{w}_i] + \lambda \sum_{k=1}^{n} w_{ki} \log w_{ki} \right]. \tag{44}$$

These distances can then be input to hierarchical clustering algorithms.

The weight values $\mathbf{W}^* = \{\mathbf{w}_i^*\}_1^N$ minimizing (44) are those that create the smallest K–nearest neighborhoods, subject to the negative entropy incentive (19). Here the size of each neighborhood is measured by the average distance to its center point $\mathbf{x}_i$, using attribute weights $\mathbf{w}_i = \{w_{ki}\}_1^n$. This is inversely related to an estimate, based on $KNN(i)$, of the probability density $p(\mathbf{x}_i \,|\, \mathbf{w}_i)$. In this sense, the solution weights minimizing (44) are chosen to maximize these probability density estimates.

The considerations concerning the value of the scale parameter λ and homotopy rate parameter α (line 6) are the same as those for the COSA1 algorithm discussed in Section 6 above. The size K chosen for the nearest neighborhoods is not critical and results are fairly stable over a wide range of values. It should be large enough to provide stable estimates of S_{ki} (42) but not too much larger than the size of the cluster containing the ith object. Setting $K \backsimeq \sqrt{N}$ is a reasonable choice, although some experimentation may be desirable after reviewing the sizes of the uncovered clusters.

8 Robust dispersion measures

Measurements of the dispersion of attribute values for sets of objects (10) (12) (42) play an important role in the COSA procedures. These dispersion measures are based on computing mean values of the inter object distances on the respective attributes. For numeric valued attributes (8), mean statistics are known to be highly sensitive to a small number of objects with unusually large values ("outliers"). Using medians as an alternative measure of central tendency eliminates this problem making the overall procedure more robust. Since robustness is an important property for any data mining method, we replace the respective mean values in (10) (12) (42) with medians for numeric valued attributes. Furthermore, for computational efficiency, (10) is approximated by

$$s_k \simeq IQR(\{x_{ik}\}_{i=1}^N)/1.35 \tag{45}$$

where IQR is the interquartile range. The divisor in (45) is the value appropriate for a normally distributed variate. (The corresponding values for a uniform and log–normal distributions are 1.12 and 1.62 respectively, so that 1.35 represents an average choice.) For categorical (nominal) attributes there is no corresponding outlier issue so that (10) (12) and (42) can be used to compute the respective attribute dispersions.

9 Interpretation

If the clustering procedure is successful in uncovering distinct groups (clusters), one would like to know whether each such group represents clustering on a subset of the attributes, and if so, to identify the relevant attribute subsets for each of the respective groups. Clustering algorithms used with COSA only report group membership (1). With this approach, there is no *explicit* attribute subset selection. However, the relative importance (relevance) of each attribute k to the clustering of each clustered group G_l is given by (22) substituting the robust (Section 8) analog of S_{kl} (12)

$$S_{kl} = \frac{1}{N_l} \sum_{i \in G_l} median\{d_{ii'k}\}_{i' \in G_l} \tag{46}$$

for numeric valued attributes.

An unnormalized first order approximation

$$I_{kl} = [S_{kl} + \varepsilon]^{-1} \tag{47}$$

can be interpreted as an *absolute* measure of the importance I_{kl} of attribute k to the clustering of solution group G_l. Here, ε^{-1} represents the maximum obtainable importance value. When computed over all objects in the data set, rather than over objects within an individual cluster, (47) evaluates to $I_k = [1 + \varepsilon]^{-1}$for all attributes, due to the normalization in (7) and (10). Thus one can interpret (47) as inversely measuring the spread of x_{ik} values within the group G_l relative to its corresponding spread over all objects. For example, a value of $I_{kl} = 4$ implies that the spread of x_{ik} values within G_l is roughly one quarter of that over all of objects in the data set. Large values of I_{kl} indicate that G_l is highly clustered on attribute x_k, whereas small values indicate the opposite. Inspection of the values of $\{I_{kl}\}_{k=1}^{n}$ for each cluster G_l, allows one to ascertain the relevant attributes contributing to the clustering of the lth group G_l. Illustrations are provided in Section 12.

10 Missing values

In many applications there are incomplete data; some of the attribute values for the objects are missing. The distance measure (34) can be modified to accommodate missing values while taking advantage of the information present in the non missing values. One simply makes the modification

$$w_k \leftarrow w_k \cdot I(x_{ik} \neq \text{missing}) \cdot I(x_{jk} \neq \text{missing}) \tag{48}$$

in (34), and then renormalizing the weights to sum to one. This assigns a weight value of zero to the kth attribute in the distance calculation if its value is missing on either object i or object j. If the two objects have no non missing values in common, they are assigned an infinite distance so that they will not be placed in the same cluster.

For the calculation of the weights (22) and (43) on the kth attribute, only non missing values of that attribute (x_k) are used to calculate S_{kl} (12) or S_{ki} (42). If in (42) object i is missing a value for x_k, or all K nearest nearest neighbors of object i are missing values of x_k , then the corresponding weight is set to zero, $w_{ki} = 0$.

11 Targeted clustering

The COSA algorithms attempt to uncover distinct groups of objects that have similar joint values on subsets of the attributes. The actual joint values of the attributes in the subset about which the objects cluster is unspecified; the attempt is to find clustering centered about any possible joint values of the attributes. This may not always be the goal; there may be preferred values on some or all of the attributes about which one would like to focus.

For example, one might have data on the spending habits of consumers in terms of amounts spent on various products or activities. The goal might be to identify groups of consumers (objects) that spend relatively large amounts on subsets of the products (attributes), and be unconcerned with those who spend moderate to small amounts. Attempting to find arbitrary clustering could obscure small but potentially interesting clusters of such high spenders. Alternatively, one might be interested in identifying clusters of low spenders, or perhaps clusters of extreme spenders who either spend excessively large or small but not moderate amounts on various items. In contrast, specific consumer research might want to focus on consumers that do in fact spend moderate amounts of money. Similarly, in gene expression data one might seek clusters of samples (objects) that have preferentially high/low or extreme expression levels on subsets of the genes (attributes). Again seeking clusters centered at arbitrary values can cause difficulty in uncovering the structure of interest, especially if it is fairly subtle.

11.1 Single target clustering

Focused or targeted clustering can be accomplished by modifying the distance definitions (7) on selected individual attributes. Let t_k be a predefined target value on the kth attribute and (x_{ik}, x_{jk}) be the corresponding respective values of objects i and j on that attribute. Define the "targeted" distance between objects (i, j) on the kth attribute as

$$d_{ijk}(t_k) = \max[d_k(x_{ik}, t_k), d_k(x_{jk}, t_k)], \tag{49}$$

where

$$d_k(x, t) = |\, x - t \,|/s_k, \tag{50}$$

with s_k given by (45) for numeric attributes, and

$$d_k(x, t) = I(x \neq t)/s_k \tag{51}$$

with s_k given by (10) for categorical (nominal) attributes. This distance (49–51) is small only if both the values of x_{ik} and x_{jk} are close to each other *and* close to the target value t_k. Using (49–51) in place of (7–9) for any attribute or set of attributes will cause the clustering algorithm, when considering groupings on those attributes, to only consider clusters near the targeted values. This can substantially reduce the cluster search space, making subtle clustering near the target values easier to uncover.

For the consumer spending data example, setting target values near the maximum data value on each attribute will cause a clustering algorithm to only seek clusters of high spenders. Similarly clusters of only high (or low) gene expressions can be sought through the same mechanism. In both cases restricting the search makes it more likely to find the targeted clusters of interest, since the algorithm will not be distracted by other perhaps more dominant (but less interesting) clustering.

11.2 Dual target clustering

Single target clustering (49–51) can be quite powerful in uncovering subtle clustering effects as will be illustrated in Section 12.1. However, for some applications it can be too restrictive. In the consumer spending problem one may be interested in clusters of "extreme" spenders, people that either spend unusually high or low amounts on sets of items. Similarly, one might be seeking clusters of samples with unusually high or low (but not moderate) gene expression levels. This type of clustering can be accomplished by using "dual target" distances

$$d_{ijk}(t_k, u_k) = \min[d_{ijk}(t_k), d_{ijk}(u_k)] \tag{52}$$

on selected attributes x_k, where $d_{ijk}(\cdot)$ is the corresponding single target distance (49). This distance (52) is small whenever x_{ik} and x_{jk} are either both close to t_k *or* both close to u_k. In the consumer spending and gene expression examples one might set t_k and u_k respectively to

values near the maximum and minimum data values of the attributes. Using (52) with COSA will cause the clustering algorithm to seek clusters of extreme attribute values, ignoring (perhaps dominant) clusters with moderate attribute values.

12 Illustrations

In the following sections we illustrate COSA on several data sets. In all the examples presented here COSA2 (Section 7) was employed with average linkage hierarchical clustering so that the resulting cluster structures can be visualized. The value of the scale parameter λ (23) was taken to be $\lambda = 0.2$ and the number of nearest neighbors K (39) was taken to be the square–root of the sample size. For the attribute importance calculations (47), ε was set to 0.05.

12.1 Simulated data

In this section we present a modest systematic investigation of the properties of COSA based clustering, and its relation to traditional approaches based on Euclidean distance. Both are applied to a series of simulated data sets of size characteristic of those produced by gene expression microarray experiments. Specifically all data sets consisted of $N = 100$ objects and $n = 10000$ attributes. To aid interpretation, the simulated clustering structure was taken to be very simple.

Each data set consisted of two groups (clusters). The first group was a random sample of 85 objects drawn form a 10000–dimensional standard normal distribution. The second group of 15 objects was also drawn from a 10000–dimensional normal distribution, but its first n_0 attributes each had a mean of $\mu = 1.5$ and standard deviation $\sigma = 0.2$. The remaining $(10000 - n_0)$ attributes of the second group each had zero mean and unit standard deviation. Thus, the population distributions of the two groups differ only on the first n_0 attributes. After generation, the pooled sample was standardized to have zero mean and unit variance on all attributes. These data thus contain a small group that exhibits clustering on only a few (n_0) attributes, together with a large non–clustered background. The purpose is to study the ability of the respective clustering approaches to uncover the second small group as a function of the number of attributes n_0 upon which it clusters. Figure 1 shows histograms of the pooled data on the first nine attributes $(n_0 > 9)$. Histograms of the other $(n_0 - 9)$ clustered attributes are similar in that they show little evidence of clustering on any of the individual marginal distributions of the attributes relevant to the clustering.

Clustering based on three distance measures are compared: squared Euclidean distance, non–targeted COSA distance, and single target COSA distance with the target on each attribute set to 95 percentile of its data distribution. Figure 2 shows average linkage dendrograms for three values of n_0: 10, 60, 150 (top to bottom) for each of the three distance measures (left to right).

For $n_0 = 10$, clustering based on targeted COSA distance readily distinguishes the small 15–object cluster from the background; Euclidean and non–targeted COSA distances are unable to do so. At $n_0 = 60$, targeted COSA dramatically distinguishes the small group, whereas non–targeted COSA is seen to barely be able to provide separation (extreme left). Euclidean distance still shows no evidence of the smaller group. With $n_0 = 150$, Euclidean distance begins to provide evidence of the smaller group, whereas both COSA distances clearly delineate it.

While simple in structure, this example provides some insight into the relative strengths of the three distance measures. It illustrates the ability of COSA based procedures to separate small groups clustering only on a tiny fraction of the total number of attributes. Non–targeted COSA was able to detect clustering on less than half of the number of attributes required by Euclidean distance. Targeting, when appropriate, is seen to be especially powerful. It was able to detect a cluster involving only 15% of the objects and 0.1% of the attributes.

The above example is especially simple in that the only structure present is the existence of the small cluster; the larger distances reflect pure unstructured noise. In actual problems this is seldom the case. Larger distances on the attributes are likely to exhibit considerable structure that may or may not be associated with clustering. Since Euclidean distance is especially

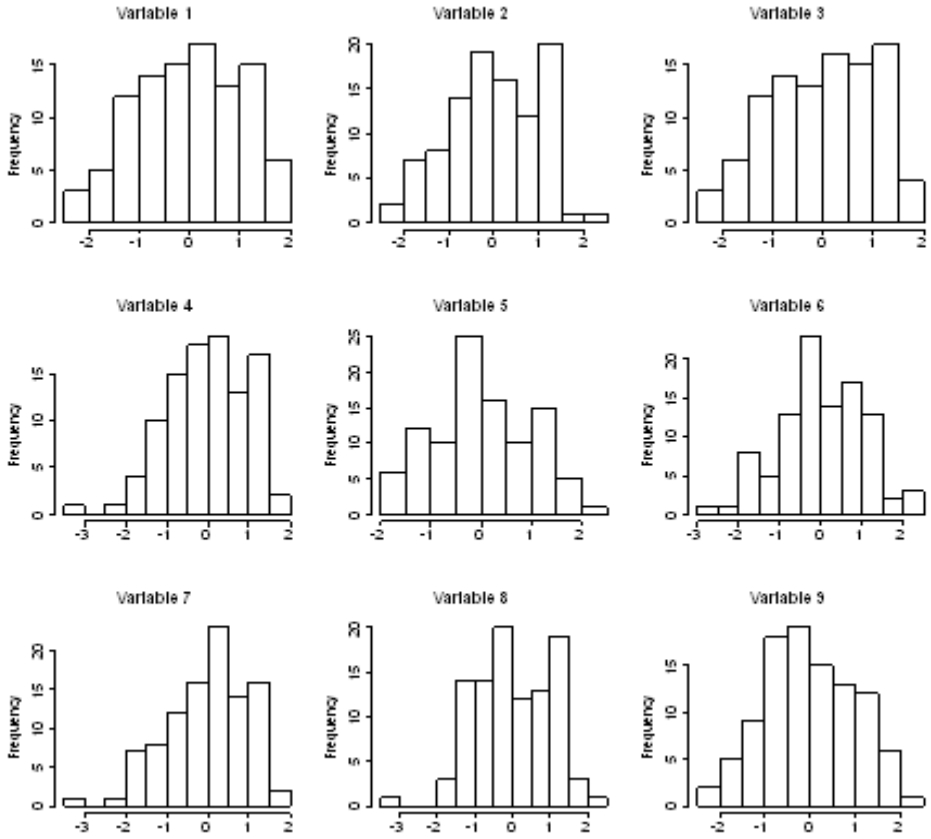

Figure 1: The distribution of the simulated clustered data on the first nine attributes. There is little evidence of obvious clustering in these marginal distributions.

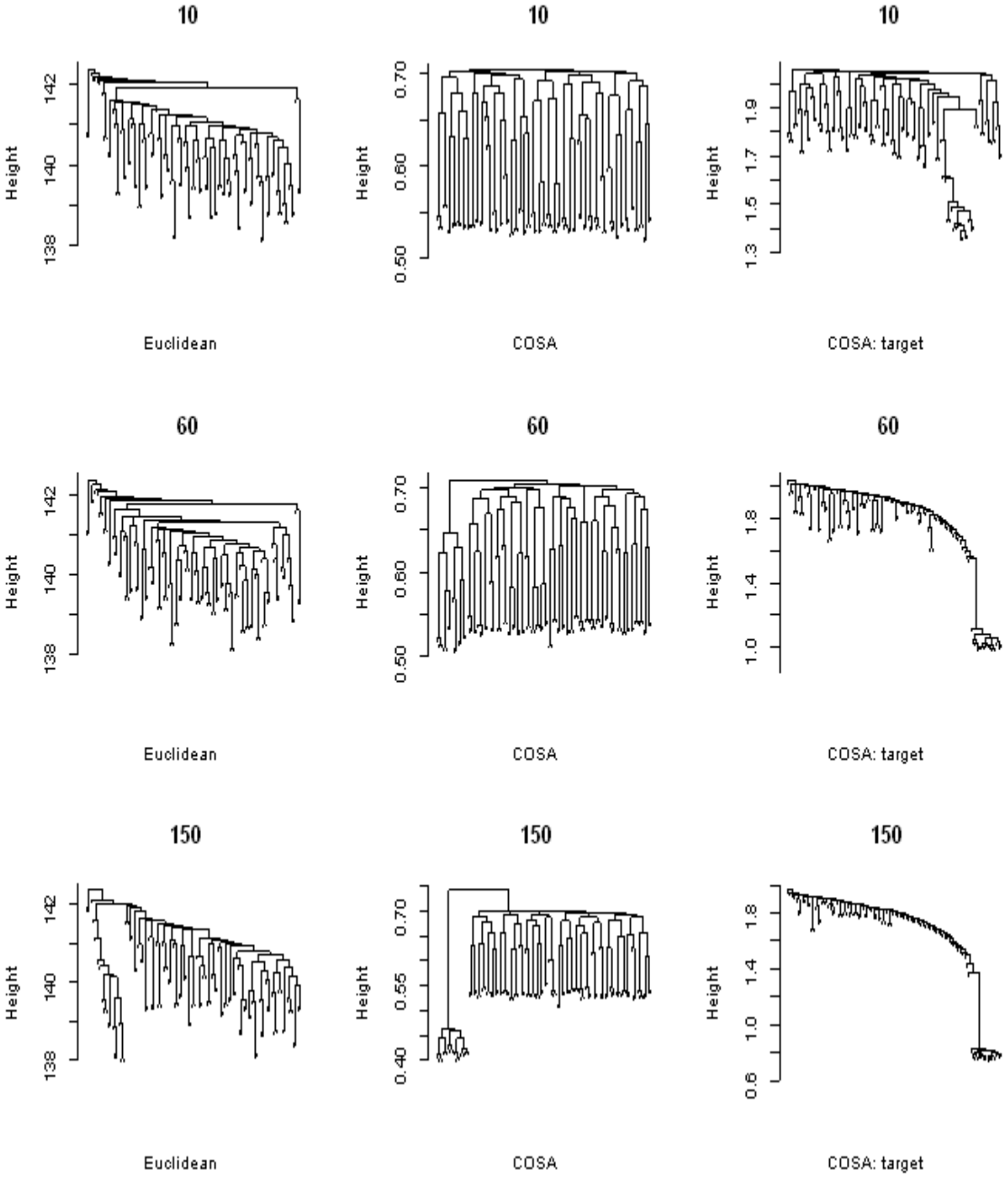

Figure 2: Average linkage dendrograms for Euclidean (left), non–targeted COSA (center), and targeted COSA (right) distances for three simulated data sets of 100 objects with 10000 attributes. Each data set consists of a small 15 object group clustering on $n_0 = 10, 60, 150$ attributes (top to bottom), nested within an unclustered background of 85 objects. Euclidean distance requires clustering on more attributes than are required by the COSA distances to detect the smaller group. Targeted COSA provides the most power in this setting.

sensitive to the larger distances on individual attributes, this large scale structure can obscure the detection of smaller groups clustering on subsets of the attributes. COSA is designed to be especially sensitive to small distances on the attributes, thereby being more sensitive to this type of structure.

The upper (black) curve in Fig. 3 shows the 100 largest attribute importance values (47) evaluated on the 15 object group detected with targeted COSA on the simulated data for $n_0 = 10$ (Fig. 2, upper right panel). The importance values for the first ten (most important) attributes are seen to be sharply higher that those for the other attributes. Their importance values range from 3.4 to 6.0 with a mean of 4.8. The population values are all 5.0 ($\sigma = 0.2$). The ten lower (green) curves represent the 100 highest attribute importance values for ten groups, each of 15 objects *randomly* selected from the data; the central red curve is their average. The importance spectrum of the actual clustered group closely coincides with those of the randomly selected groups except for the ten highest importance values. These ten attributes that are estimated to be the most important turn out to be $x_1, ..., x_{10}$, the ones actually relevant to forming the small cluster in the generating population.

The above example demonstrates the sensitivity of COSA based clustering in uncovering small groups that cluster only on very small subsets of the attributes. A potential worry is that such groupings may exist with sufficient strength in random data, sampled from an unstructured population, so that they will also be detected by COSA. The examples shown in Fig. 4 indicate that this is not likely. Shown are nine average linkage dendrograms resulting from applying single target COSA to nine different data sets of 100 objects, each randomly sampled from a 10000–dimensional standard normal distribution. As can be seen, there are no obvious indications of clustering in these plots. The corresponding dendrograms based on dual target and non–targeted COSA (not shown) are similar in that they also show no obvious clustering.

12.2 mRNA relative abundance data

This data set consists of $n = 6221$ mRNA relative abundance estimates derived from gene expression data (attributes), with $N = 213$ samples (objects). The data are an agglomeration of samples from 12 experiments derived from nine studies (Aach, Rindone, and Church 2000). Attributes with more than half of their values missing were deleted from the analysis leaving 6141 attributes still containing many missing values.

Figure 5 displays the average linkage dendrogram obtained from squared Euclidean distance on the standardized attributes. Substantial clustering is apparent. Five distinct groups that each contain ten or more objects can be identified. These are delineated in the top panel of Fig. 7. Figure 6 shows the corresponding dendrogram based on (standard) COSA distance. With COSA, the separation into distinct clusters is seen to be much sharper, and at least nine distinct groups (containing more than 10 objects) can be identified. These are delineated in the bottom panel of Fig. 7.

Each of the Euclidean clusters (Fig. 7, top panel) uniquely contain all of the objects (samples) arising from five of the 12 experiments. These are identified in Table 3 with the delineated clusters labeled sequentially from left to right. Unsupervised Euclidean distance clustering was able to separate these five experiments from the rest of the data in the absence of an experiment label.

Eight of the nine clusters identified in the COSA dendrogram (Fig. 7, bottom panel) contain objects (samples) from unique experiments. These are identified in Table 4 with the COSA clusters in Fig. 7 labeled sequentially from left to right. COSA clusters 3, 6, 7, and 8 contain all of the objects from each of the corresponding experiments. Clusters 1 and 2 partition all of the *Hol* experimental samples into two distinct groups, whereas clusters 4 and 5 similarly divide all of the samples of the *Spe_cdc* experiment. Cluster 9 is the only impure one, containing all of the *Der_duix* samples and a few samples from other experiments as well.

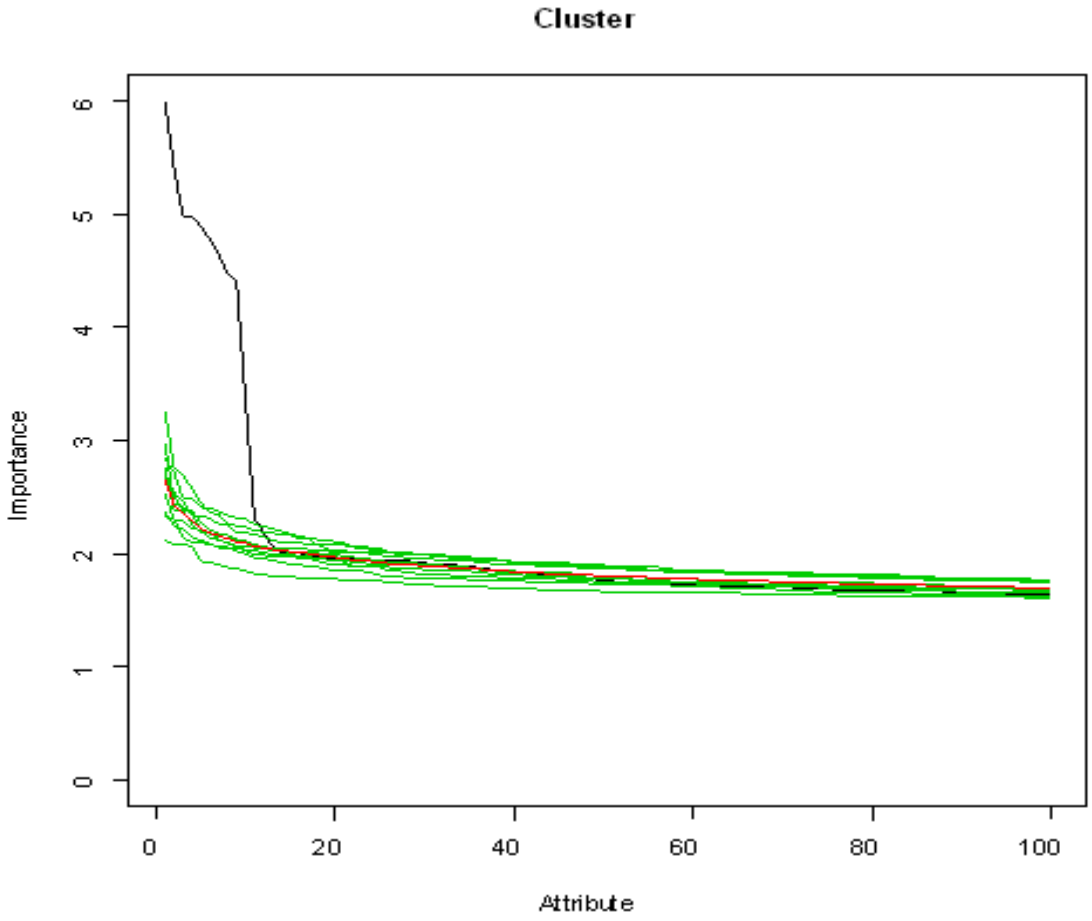

Figure 3: The 100 largest attribute importance values for the 15 object group detected by targeted COSA on the simulated data for $n_0 = 10$ (upper black curve). The lower (green) curves represent the 100 highest attribute importance values for ten groups, each of 15 objects *randomly* selected from the data. The central (red) curve is the average of the green curves. The detected group shows strong evidence of clustering on 10 attributes, with little or no evidence of clustering on the remaining 9990 attributes.

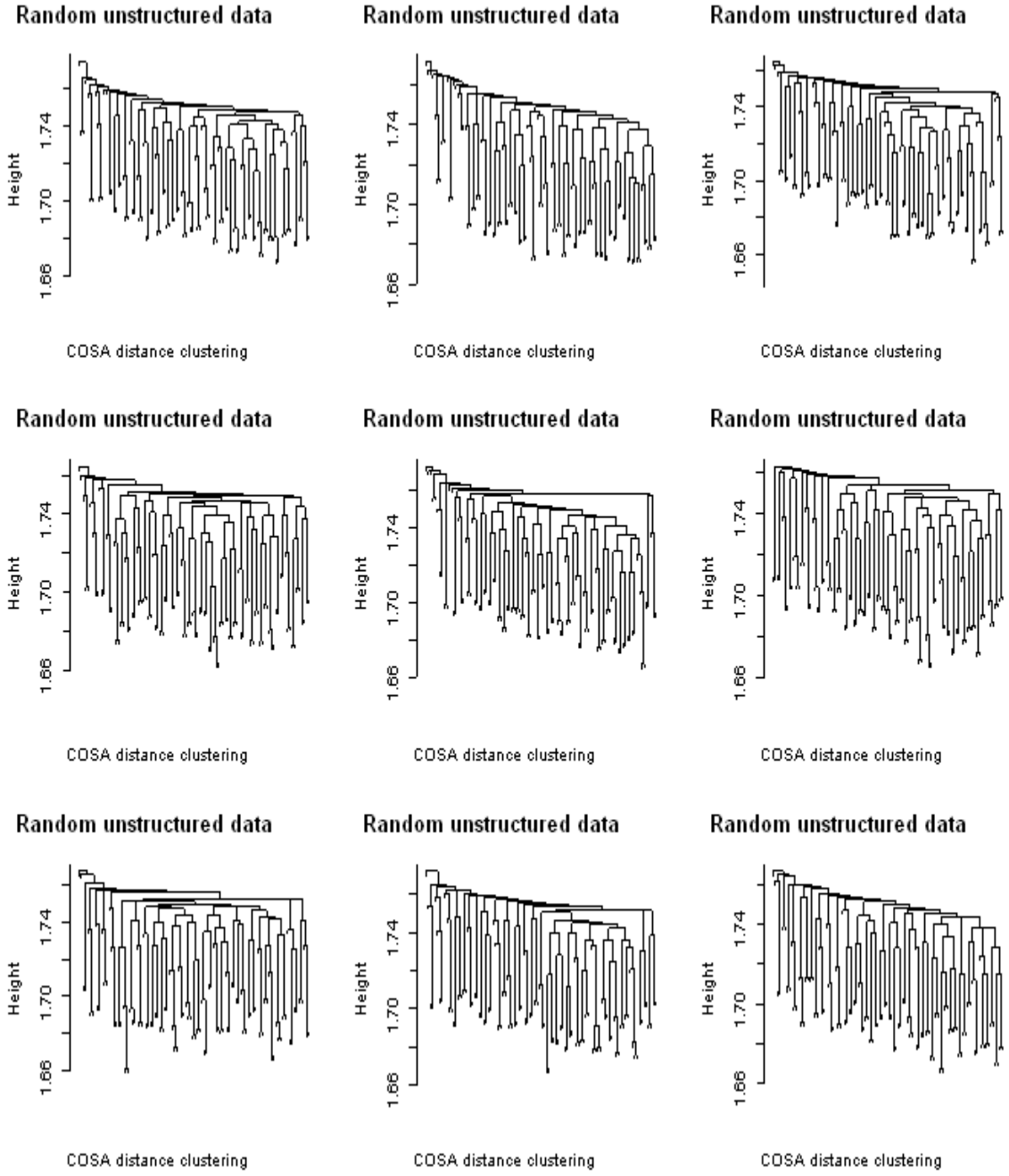

Figure 4: Average linkage dendrograms resulting from applying single target COSA distance to nine different data sets of 100 objects randomly drawn from a 10000–dimensional standard normal distribution. No indications of obvious clustering appears in any of these plots.

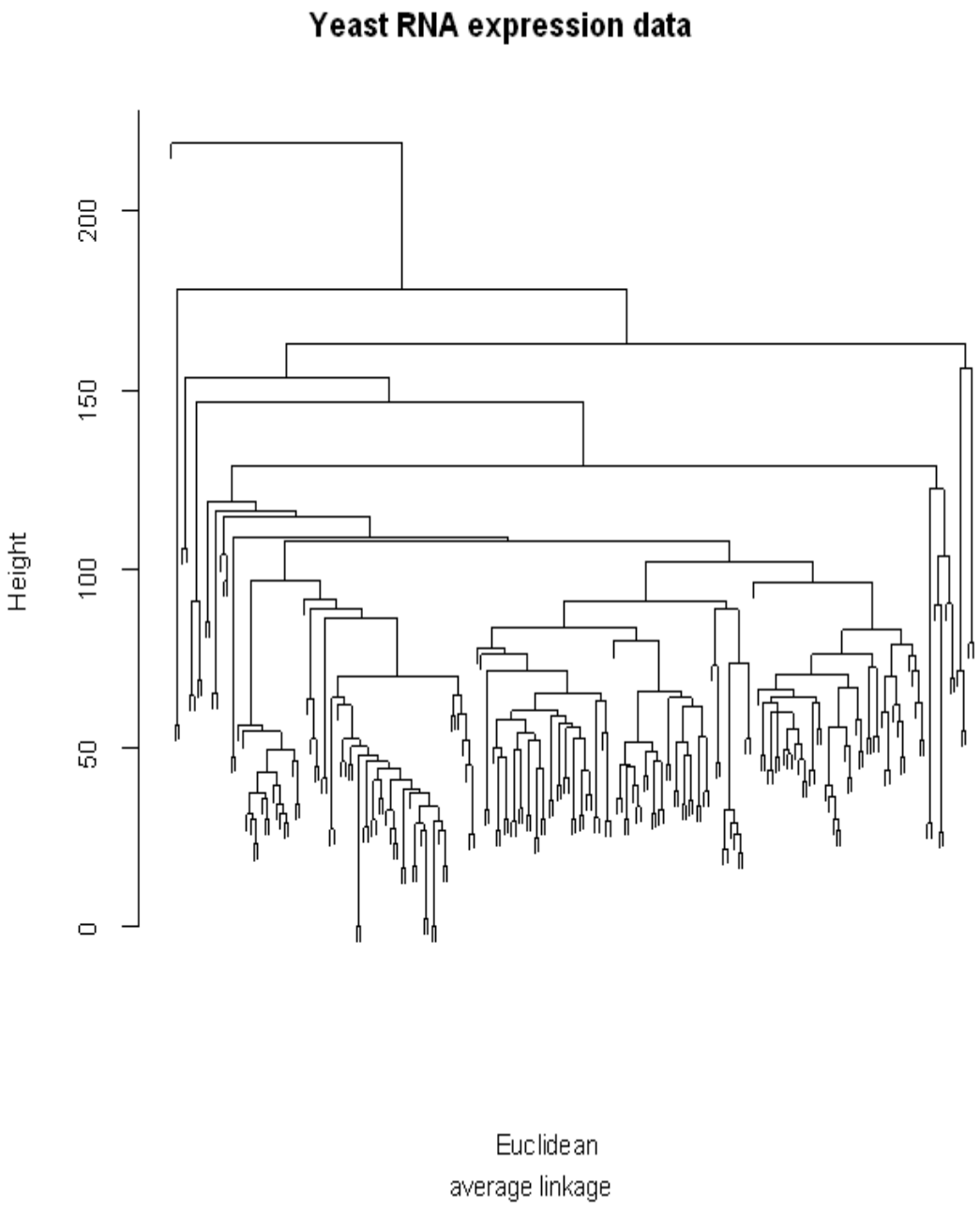

Figure 5: Average linkage dendrogram based on Euclidean distance for the yeast mRNA relative abundance data. Substantial clustering is indicated.

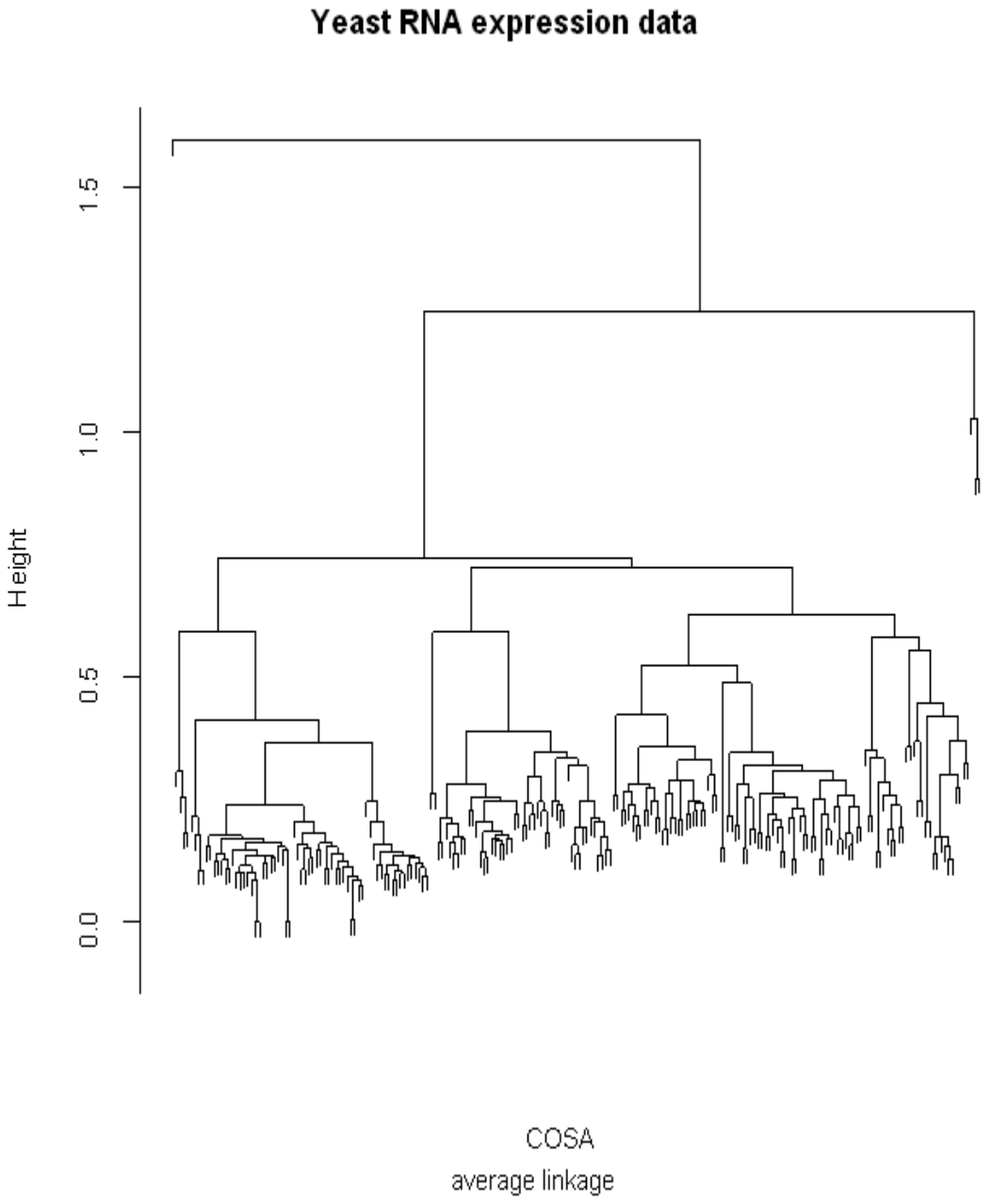

Figure 6: Average linkage dendrogram based on COSA distance for the yeast mRNA relative abundance data. Very sharp clustering is apparent.

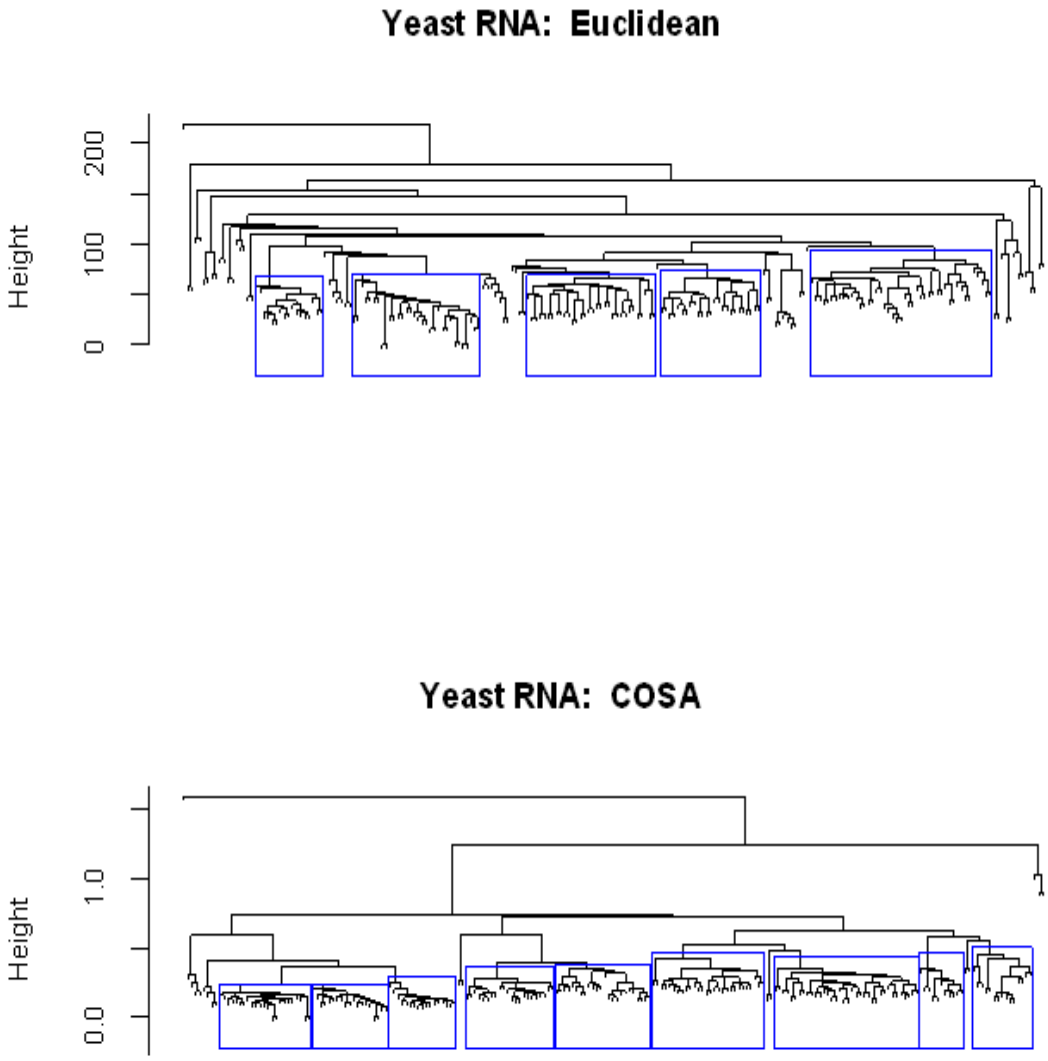

Figure 7: Average linkage dendrograms based on Euclidean (top) and COSA (bottom) distance for the yeast mRNA relative abundance data, with clustered groups involving ten or more objects delineated.

Table 3

Experiments comprising each of the Euclidean distance clusters for the yeast mRNA relative abundance data.

Cluster:	1	2	3	4	5
Exp:	*Cho*	*Hol*	*Spe_alpha*	*Spe_elut*	*Spe_cdc*

Table 4

Experiments comprising each of the COSA distance clusters for the yeast mRNA relative abundance data.

Cluster:	1	2	3	4	5
Exp:	*Hol*	*Hol*	*Cho*	*Spe_cdc*	*Spe_cdc*
Cluster:	6		7	8	9
Exp:	*Spe_elut*		*Spe_alpha*	*Chu*	*Der_diux* (+)

The results from COSA clustering suggest that the *Hol* and *Spe_ cdc* experiments each partition into two distinct groups of similar size. This is not evident from Euclidean distance based clustering.

Figure 8 illustrates the attribute importance values (upper black curves) for the two *Hol* groups. The lower (green) curves are the corresponding ordered attribute importances for same sized groups randomly selected from the whole data set. Both of the *Hol* subgroups strongly cluster on a relatively small fraction of all of the attributes. The concentration is somewhat sharper for the first (left) group. The attribute subsets on which the two groups strongly cluster are not identical, but substantially overlap. There were 41 common attributes among the 100 most relevant for each group.

Euclidean distance based clustering was able to partition five of the six experiments that contain more ten or more samples (objects) into separate groups. (The other six experiments contained less than ten samples.) COSA clustering (more sharply) separated all six of these experiments, (with a contaminated seventh cluster), and in addition was able to detect strong clustering structure *within* two of them.

12.3 Medical data

These data were collected at the Leiden Cytology and Pathology Laboratory (see Meulman, Zeppa, Boon, & Rietveld 1992). They consist of $n = 11$ manually observed features (attributes) of cells taken from pap smears of $N = 242$ patients (objects) with cervical cancer or its precursor lesions (dysplasias). Attributes $(x_1, ..., x_4)$ and $(x_6, ..., x_8)$ are abnormality ratings by a pathologist of various aspects of each cell. The ratings range from 1 (normal) to 4 (very abnormal). Most of these attributes have only three distinct values (normal ratings were rare) so they were treated as being categorical. The remaining four features $(x_5, x_9, x_{10}, x_{11})$ are numerical (counts) with many distinct values.

The strongest clustering was revealed by using dual target distance (52), with the targets set respectively to the 5 and 95 percentiles of the data distribution on each numeric attribute. There were no targets specified for the categorical attributes. Figure 9 shows the resulting average–linkage dendrogram. These data are seen to partition into nine fairly distinct groups, containing ten or more objects, delineated by the corresponding rectangles. Moderate additional clustering within some of these groups is also indicated.

The attribute importances (47) for each of these nine groups are plotted (on a square root scale) in Fig. 10. The groups are displayed in their dendrogram (Fig. 9, left to right) order. All of the groups exhibit very strong clustering on one to three attributes, with some groups

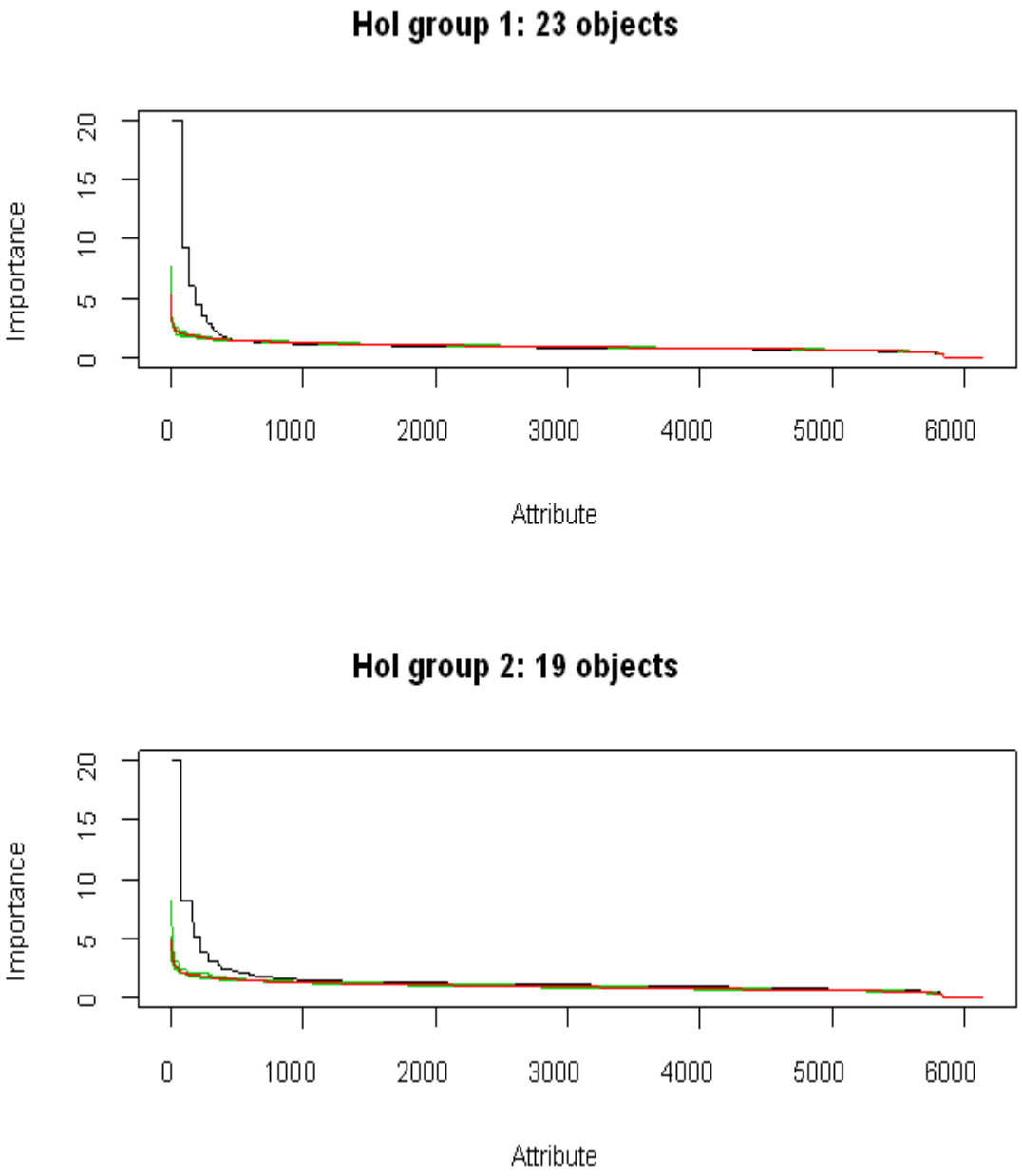

Figure 8: Ordered attribute importances for each of the two *Hol* clusters. The lower (green) curves represent the corresponding ordered importances for randomly selected groups of the same size. The central (red) curve is their average. Both of these groups exhibit strong clustering only on a small subset of the attributes (genes).

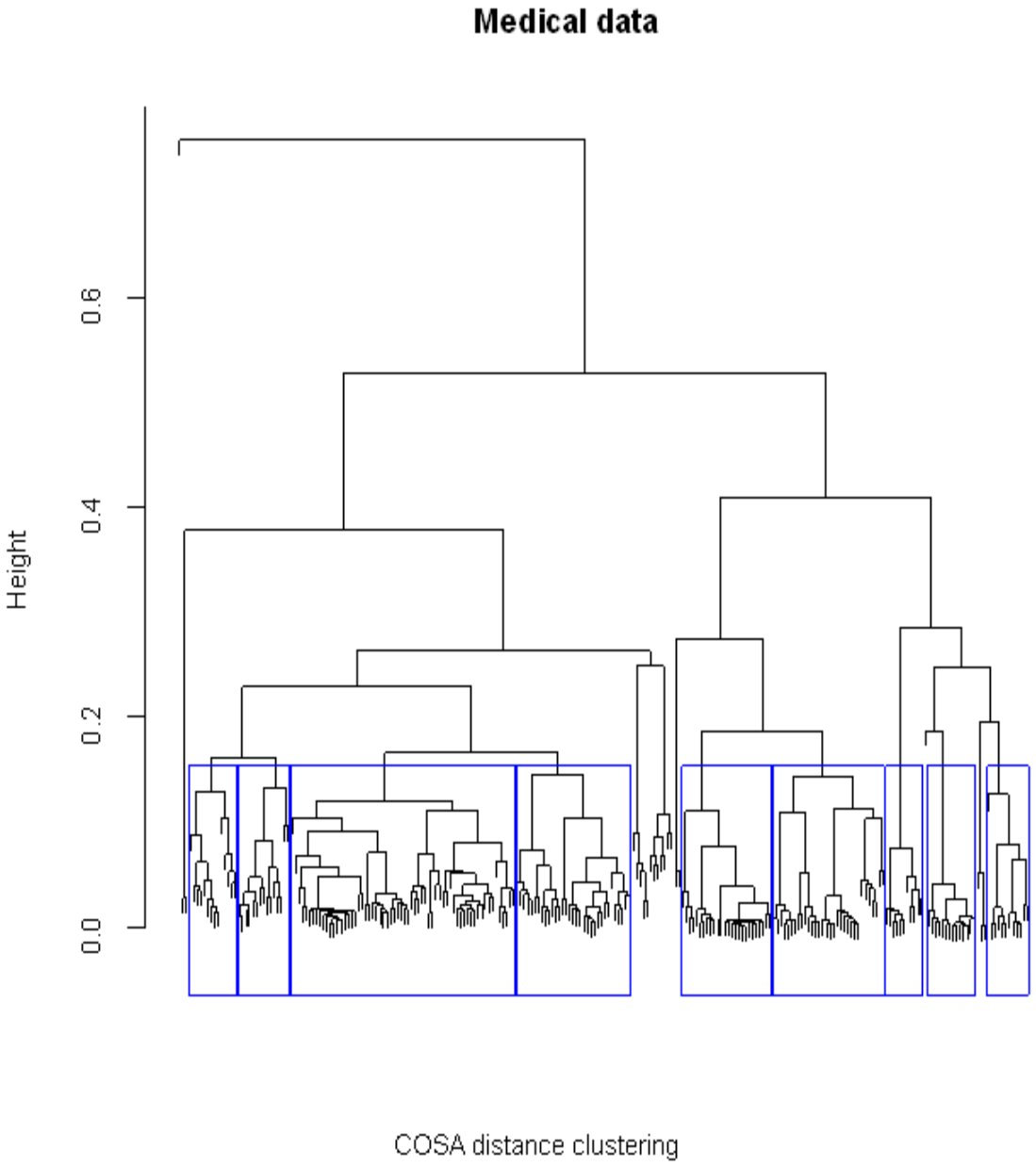

Figure 9: COSA average–linkage dendrogram for the medical data using high/low dual target dissimilarities on the numeric variables. These data are seen to partition into nine fairly well separated groups of more than ten objects each as delineated by the rectangles.

showing moderately strong clustering on a few other attributes as well. Each group is seen to cluster on a different small subset of the attributes, with some overlap among the subsets.

In addition to the eleven cell features taken from their pap smears, each patient was diagnosed in a subsequently performed biopsy. Each of these histological diagnoses was assigned a numerical score, with values ranging from one to five, reflecting the severity of the dysplasia (mild, moderate, or severe) or cervical carcinoma (in situ, or invasive). Figure 11 shows the distribution (boxplot) of these score values for patients assigned to each of the nine clustered groups identified in Fig. 9 (left to right) and shown in Fig. 10. The labels along the abscissa show the median of the index values within each of the respective groups. Each of the uncovered clusters, based only on the cell features, is seen to correspond to a relatively narrow range of increasing score values, indicating a substantial relationship between the COSA group membership and the severity of the diagnosis, with a clear separation between carcinoma (severity scores 4 and 5) and dysplasia (severity scores1, 2 and 3).

13 Discussion

COSA can be viewed as an enhancement to distance based clustering methods enabling them to uncover groups of objects that have preferentially close values on different, possibly overlapping, subsets of the attributes. There do not appear to be other distance based methods directly focused on this goal. There are however non distance based modeling methods that have been proposed for this purpose.

The one closest in spirit is product density mixture modeling (AutoClass – Cheesman and Stutz 1996, see also Banfield and Raftery 1993). The joint distribution of the attribute values is modeled by a mixture of parameterized component densities. Each component in the mixture is taken to be a product of individual probability densities on each of the attributes. Prior probability distributions are placed all model parameter values and a heuristic search strategy is used to attempt to maximize posterior probability on the data. Each of the components in the resulting solution is considered to be a "soft" cluster.

As with COSA, the (posterior probability) criteria being optimized by these methods are highly non convex functions of their parameters and avoiding convergence to distinctly inferior local optima is a problem (see Ghosh and Chinnaiyan 2002). Furthermore for large data sets, such as those derived from gene expression microarrays, the very large number of associated model parameters causes severe computational and statistical estimation difficulties. Therefore, specialized preprocessing and screening procedures are required to substantially reduce the size of the problem to manageable proportion. Also, experimenting with various data transformations is often required in an attempt to bring the data into conformity with the parametric model (Yeung, Fraley, Murua, Raftery and Ruzzo 2001). COSA is distinguished from these methods by its nonparametric formulation and computational feasibility on large data sets, thereby reducing or eliminating dependence on customized preprocessing and screening procedures. It can be used with hierarchical clustering methods, and it employs a search strategy using a particular homotopy technique in an attempt to avoid distinctly suboptimal solutions.

Motivated by gene expression microarray data, several recent techniques have been proposed to uncover clustering by directly modeling the (numeric) data matrix $\mathbf{X} = [x_{ij}] \in R^{N \times n}$ by additive decompositions. Each additive term is interpreted as a cluster. Plaid models (Lazzeroni and Owen 2000) treat the objects and attributes symmetrically. The data matrix is represented by an expansion analogous to the singular value decomposition. The components of the singular vectors for each term ("layer") in the expansion are restricted to the two values $\{0, 1\}$. A value 1 (0) for the ith component of a left singular vector indicates that the corresponding ith row of the data matrix does (does not) contribute to the clustering represented by that layer. Similarly, a value 1 (0) for the jth component of a right singular vector indicates that the corresponding column does (does not) contribute. Each layer is interpreted as modeling the data matrix after subtracting the contributions of all previous layers. Gene shaving (Hastie, Tibshirani, Eisen, Brown, Ross, Scherf, Weinstein, Alizadeh, Staudt and Botstein 2000) seeks to decompose the

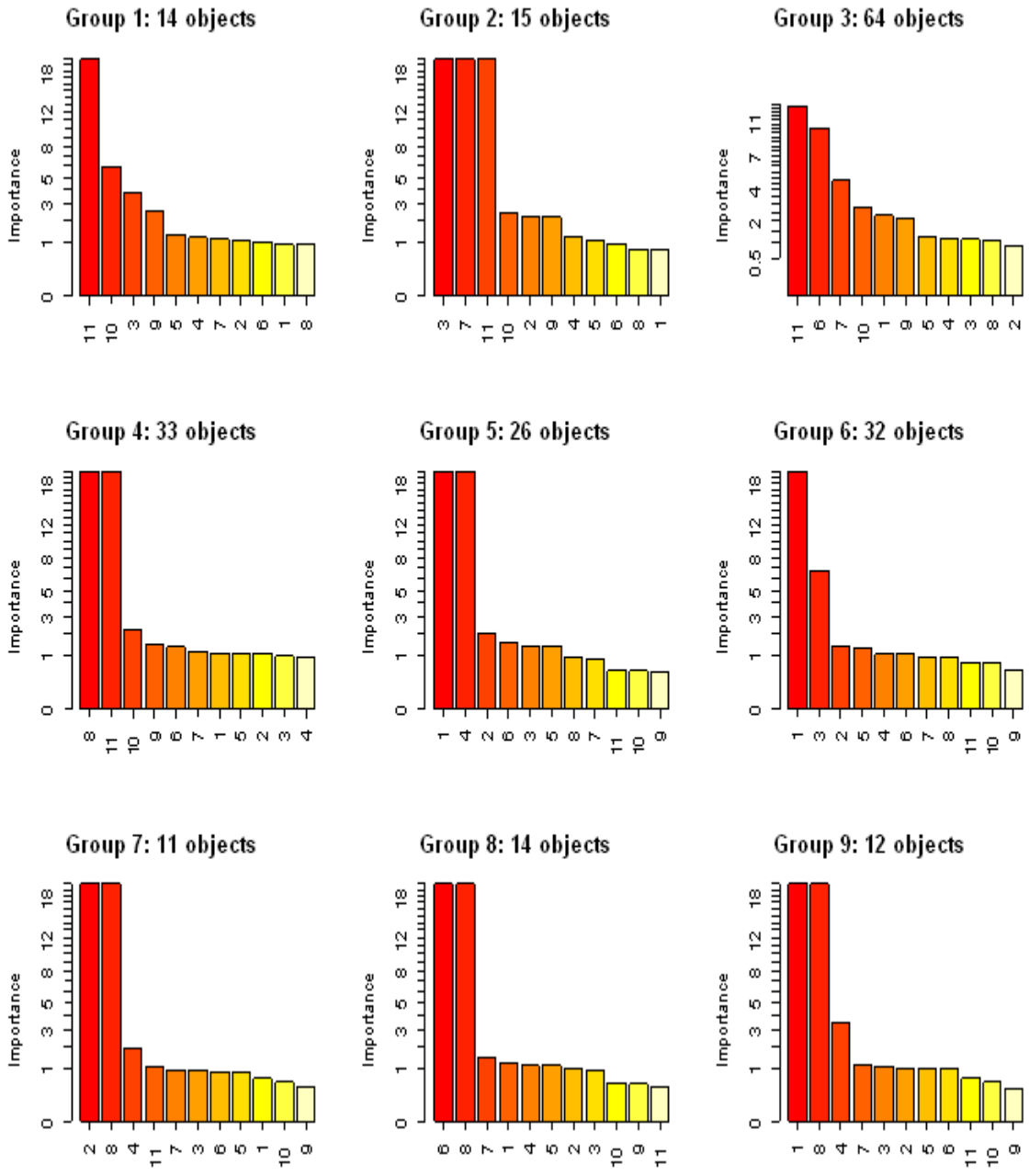

Figure 10: Attribute importances for each of the nine groups uncovered in the medical data set shown on a square–root scale. Each of these groups tend to cluster on a relatively small subset of the eleven attributes.

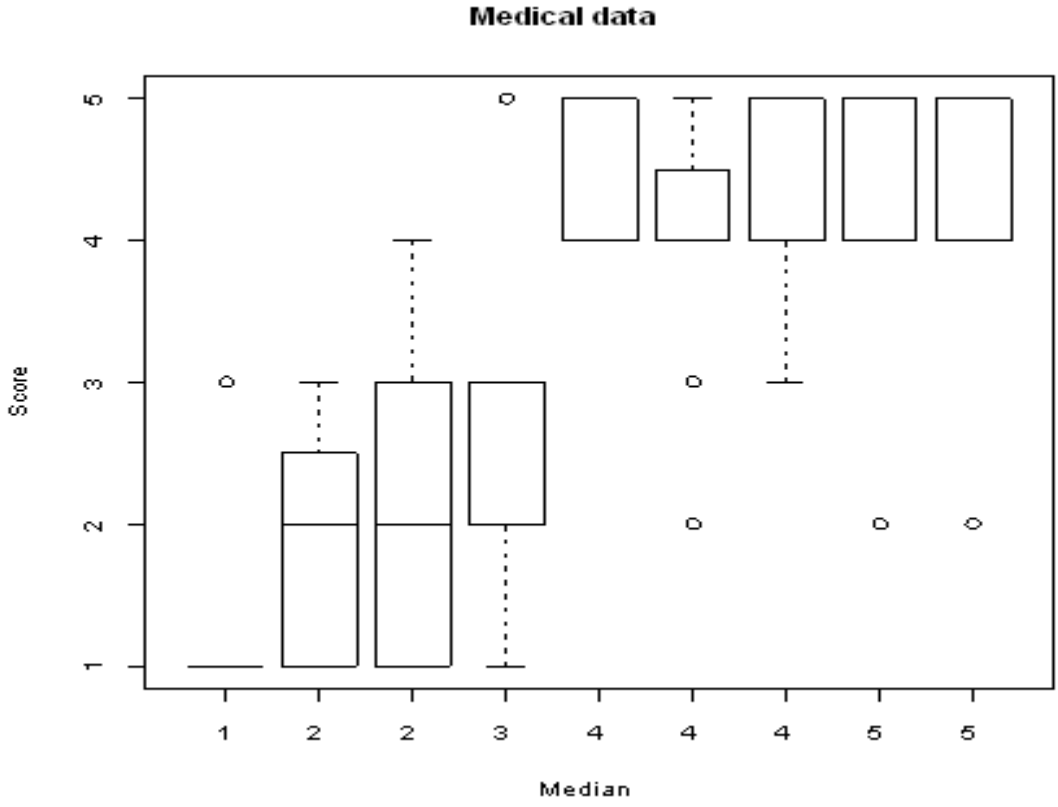

Figure 11: Distribution of diagnosis indices (from mild dysplasia = 1 to invasive carcinoma = 5) within each of the nine clusters delineated (left to right) in Fig. 9. The median index value for each group is shown along the abscissa. Each respective group corresponds to a relatively narrow range of increasing index values, indicating a relationship between cluster membership and disease severity.

$N \times n$ data matrix into a set of smaller $N_k \times n$ matrices ($1 \leq k \leq K$, $N_k << N$) such that within each the components of the row mean vector (averaged over the columns) exhibit high variance. The rows within each such matrix are interpreted as clusters.

Although by no means the same, the underlying goals of all of these methods are somewhat similar. As with all such methods, a major component is the particular heuristic search strategy employed. Even with similar (or the same) goals, different search strategies have the potential to reach quite different solutions representing different clustering structures, many of which may be interesting and useful. The particular characteristics of the COSA method proposed in this paper include the "crisp" or "hard" clustering of objects on possibly overlapping subsets of attributes, the use of targets anywhere within the domain of each attribute to focus the search on particular types of "interesting" structure, and as noted above, a homotopy technique based on weighted inverse exponential distance to avoid suboptimal local solutions. We conjecture that the use of the latter is crucial in finding the weights for the attributes that define the subsets for each separate cluster of objects. The COSA technique can be used in conjunction with a wide variety of (distance based) clustering algorithms, including hierarchical methods, each employing its own particular encoder search strategy. As with any data analytic procedure, the validity and usefulness of the output of different clustering methods can only be evaluated by the user in the context of each particular application.

14 Acknowledgments

The work of Jerome H. Friedman was partially supported by the Department of Energy under contract DE–AC03–76SF00515, and by the National Science Foundation under grant DMS–97–64431.

References

[1] Aach, J., Rindone, W., and Church, G. M. (2000). Systematic management and analysis of yeast gene expression data. *Genome Research* **10**, 431-44

[2] Arabie, P., Hubert, L. J., and De Soete, G. (Eds.), (1996). *Clustering and Classification.* River Edge, New Jersey: World Scientific Publ.

[3] Banfield, J. D. and Raftery, A. E. (1993). Model–based Gaussian and non–Gaussian clustering. *Biometrics* **49**, 803-821.

[4] Brusco, M. J. and Cradit, J. D. (2001). A Variable Selection Heuristic for K-Means Clustering. *Psychometrika* **66**, 249-270.

[5] Cheesman, P. and Stutz, J. (1996). Bayesian classification (AutoClass): Theory and results. In *Advances in Knowledge Discovery and Data Mining*, Fayyad, Piatetsky–Shapiro, Smyth, and Uthurusamy, eds. Cambridge, MA: AAAI/MIT Press 153–180.

[6] DeSarbo, W. S., Carroll, J. D., Clarck, L. A., and Green, P. E. (1984). Synthesized Clustering: A method for amalgamating clustering bases with differential weighting of variables. *Psychometrika* **49**, 57-78.

[7] De Soete, G. (1986). Optimal variable weighting for ultrametric and additive tree clustering. *Quality and Quantity* **20**, 169-180.

[8] De Soete, G. (1988). OVWTRE: A program for optimal variable weighting for ultrametric and additive tree fitting. *Journal of Classification* **5**, 101-104.

[9] De Soete, G., DeSarbo, W. S., and Carroll, J. D. (1985). Optimal variable weighting for hierarchical clustering: An alternating least-squares algorithm. *Journal of Classification* **2**, 173-192.

[10] Fowlkes, B. E., Gnanadesikan, R., and Kettenring, J. R.. (1988). Variable selection in clustering. *Journal of Classification* **5**, 205-228.5.

[11] Ghosh, D. and Chinnaiyan, A. M. (2002). Mixture modeling of gene expression data from microarray experiments. *Bioinformatics* **18**, 275-286.

[12] Gnanadesikan, R., Kettenring, J. R., and Tsao, S. L. (1995). Weighting and selection of variables for cluster analysis. *Journal of Classification* **12**, 113-136.

[13] Gordon, A. (1999). *Classification (2nd edition).* Chapman & Hall/CRC press, London.

[14] Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics* **27**, 857-871.

[15] Hansen, P., and Jaumard, B. (1997). Cluster analysis and mathematical programming. *Mathematical Programming* **79**, 191-215.

[16] Hartigan, J. A. (1975). *Clustering Algorithms.* Wiley, New York.

[17] Hastie, T., Tibshirani, R., Eisen, M., Brown, P., Ross, D., Scherf, U., Weinstein, J., Alizadeh, A., Staudt, L., and Botstein, D. (2000). Gene shaving: a new class of clustering methods for expression arrays. Technical report, Stanford University, Statistics.

[18] Hubert, L. J., Arabie, P., and Meulman, J. J. (2001). *Combinatorial Data Analysis: Optimization by Dynamic Programming.* Philadelphia: SIAM.

[19] Jain, A. and Dubes, R. (1988). *Algorithms for Clustering Data.* Prentice–Hall.

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

[21] Lazzeroni, L. and Owen, A. (2000). Plaid models for gene expression data. Technical report, Stanford University, Statistics.

[22] Meulman, J. J., Zeppa, P., Boon, M. E., and Rietveld, W. J. (1992). Prediction of various grades of cervical preneoplasia and neoplasia on plastic embedded cytobrush samples: discriminant analysis with qualitative and quantitative predictors. *Analytical and Quantitative Cytology and Histology* **14**, 60-72.

[23] Milligan, G. W. (1989). A validation study of a variable-weighting algorithm for cluster analysis. *Journal of Classification* **6**, 53-71.

[24] Mirkin, B. G. (1996). *Mathematical Classification and Clustering.* Boston: Kluwer Academic Publishers.

[25] Späth, H. (1980). *Cluster Analysis Algorithms.* Chicester, U.K.: Ellis Horwood.

[26] Van Buuren, S. and Heiser, W. J (1989). Clustering n objects into k groups under optimal scaling of variables. *Psychometrika* **54**, 699-706

[27] Van Os, B. J. (2001). *Dynamic programming for partitioning in multivariate data analysis.* Unpublished PhD thesis. Leiden University: Dept. of Data Theory.

[28] Yeung, K. Y., Fraley, C., Murua, A., Raftery, A. E. and Ruzzo, W. L. (2001). Model–based clustering and data transformations for gene expression data. Proceedings: *The Third Georgia Tech–Emory International Conference on Bioinformatics* (to appear).